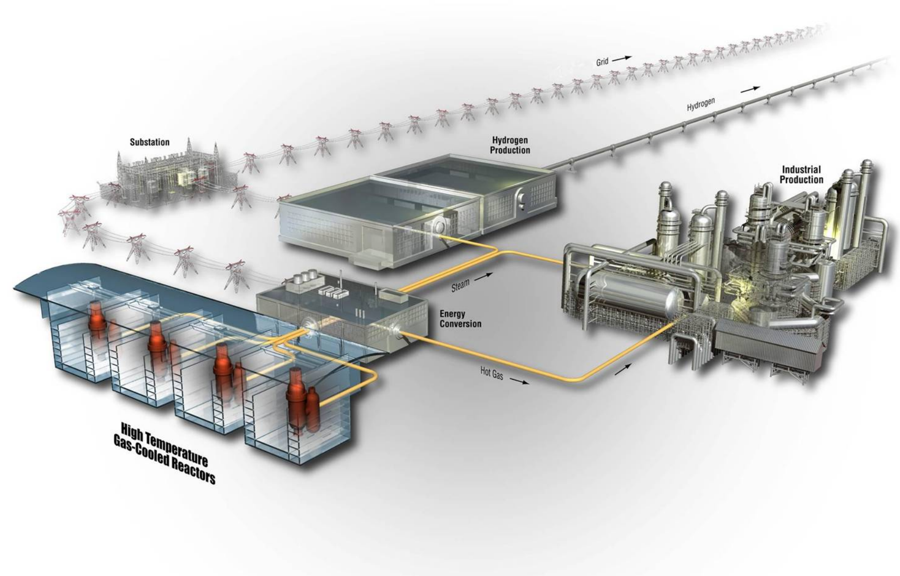
INL/EXT-12-27594

NDMAS System and Process Description

Larry Hull

October 2012



Idaho National Laboratory

INL/EXT-12-27594

NDMAS System and Process Description

Larry Hull

October 2012

Idaho National Laboratory
VHTR Program
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

VHTR Program

NDMAS System and Process Description

INL/EXT-12-27594

October 2012

Approved by:

Larry Hull
Author

10/31/2012
Date

Jeffery Einerson
NDMAS Project Lead

10-31-2012
Date

David R. Jensen
Quality Assurance

10/31/12
Date

Diane V. Croson
VHTR TDO Deputy Director

10/31/12
Date

ABSTRACT

Experimental data generated by the Very High Temperature Reactor Program need to be more available to users in the form of data tables on Web pages that can be downloaded to Excel or in delimited text formats that can be used directly for input to analysis and simulation codes, statistical packages, and graphics software. One solution that can provide current and future researchers with direct access to the data they need, while complying with records management requirements, is the Nuclear Data Management and Analysis System (NDMAS). This report describes the NDMAS system and its components, defines roles and responsibilities, describes the functions the system performs, describes the internal processes the NDMAS team uses to carry out the mission, and describes the hardware and software used to meet Very High Temperature Reactor Program needs.

CONTENTS

FIGURES

TABLES

ACRONYMS

AGC	Advanced Graphite Capsule
AGR	Advanced Gas Reactor
ATR	Advanced Test Reactor
EBI	Enterprise Business Intelligence
CDCS	capsule distributed control system
CR	change request (work item)
EDMS	Electronic Document Management System
ETL	extract, transform, and load
FPMS	fission product monitoring system
HTM	High Temperature Materials
HTML	hypertext markup language
INL	Idaho National Laboratory
NDMAS	Nuclear Data Management and Analysis System
NGNP	Next Generation Nuclear Plant
NQA	Nuclear Quality Assurance
OLAP	online analytical processing
PIE	post-irradiation examination
QA	quality assurance
RDAS	reactor data acquisition system
SQA	software quality assurance
SQL	structured query language
TFS	Team Foundation Server
TDO	Technology Development Office
VHTR	very high temperature reactor

NDMAS System and Process Description

1. INTRODUCTION

1.1 Purpose of Very High Temperature Reactor Program

Research and development activities to support design and licensing of the very high temperature reactor (VHTR) involve collecting data to determine how nuclear fuel and other reactor structural materials react to irradiation at the high temperatures (>900°C) expected in a VHTR. Materials for use in reactor cores (fuel and graphite) are fabricated and characterized for physical, chemical, thermal, and mechanical properties. Fuel and material specimens are then placed in the Advanced Test Reactor (ATR) and irradiated to the specified neutron dose. Reactor operating conditions and gaseous fission products are monitored during irradiation experiments and dose and thermal calculations are performed to characterize conditions during irradiation. Post-irradiation examination (PIE) is performed to quantify changes in material properties caused by irradiation. Fuel fabrication process data are gathered to support design of a full-scale fuel manufacturing facility and the fuel property data provide a basis for fuel specifications. Other data collection projects study the performance of materials for use in high-temperature heat exchangers and collect data to validate thermal-hydraulic codes. Data may come from existing sources, experiments conducted by the program, and experiments conducted by universities, other laboratories, and private companies.

1.2 Purpose of NDMAS

The U.S. Department of Energy has made a significant investment in collecting VHTR Program data from research conducted by Idaho National Laboratory (INL), universities, other laboratories, and private companies. This research will continue to generate a large amount of data from a variety of sources over a period of many years. Managing this data to retain data integrity and availability presents a significant challenge.

1.2.1 Records Management

The experimental data generated by the VHTR Program are considered records and must be managed following the PLN-3319, “Records Management Plan for the VHTR Technology Development Office Program.” These records are then sent to the Records Service Center, where they are uploaded into the INL Electronic Document Management System (EDMS) in native file format. As the name suggests, data that are loaded into EDMS as electronic documents are still treated as documents, not as data. These data need to be more available to users in the form of data tables on Web pages that can be downloaded to Excel or in delimited text formats that can be used directly for input to analysis and simulation codes, statistical packages, and graphics software. The VHTR Program therefore needs a solution that can provide current and future researchers with direct access to the data they need, while complying with records management requirements.

PLN-3319 allows for files to be stored in a satellite file location, which is a location different from the Records Service Center or EDMS, to support retrieval needs. The Nuclear Data Management and Analysis System (NDMAS) is designated as a satellite file location for storage of data records generated by the VHTR Program. Submitting data to NDMAS meets the data stream owner obligations to comply with PLN-3319. NDMAS makes data records more available to end users by capturing data from a variety of formats, transforming the data into a machine-useable format, and making the data available on the Web where it can be directly and easily accessed by end-user analysis tools. This allows researchers rapid access to data for analysis and interpretation rather than reformatting data from electronic documents into useable form.

NDMAS supports data qualification, stores data in a controlled and secure electronic environment, identifies the qualification status of data, provides data analysis and modeling products, and makes data available for use by the program. It is Web-based, so program members can access the system and review the data, obtain analysis results, including statistics and graphics, create slide presentations, and download data for advanced analysis. By performing these roles, the NDMAS ensures the correct data are used by the program and that data of known quality will be available to support licensing in the future.

1.2.2 Scope

NDMAS supports the scope of the VHTR Program as laid out in the PLN-2494, “VHTR Technology Development Office Program Management Plan.” Other data provided by VHTR Program data generators relevant to the mission of the VHTR Program will also be added to the database at the request of the program research staff.

NDMAS supports all VHTR Program data collection projects. Requirements for data storage, data retrieval, and Web page content are derived from the data collection activities. As the VHTR Program evolves over time, the data collection activities will change and grow. The NDMAS system must have flexibility to change and grow over time as the data streams and system supports change.

1.3 Objective

This report provides an overview of the NDMAS system and its components, defines roles and responsibilities, describes the functions performed by the system, describes the internal processes the NDMAS team uses to carry out the mission, and describes the hardware and software used to accomplish the VHTR Program needs. This report only briefly describes data analysis and delivery of data to the Web; the report, *NGNP Data Management and Analysis System Analysis and Web Delivery Capabilities* (Gentillon et al. 2011) provides a more detailed description of those capabilities.

2. PROCESS MODEL

This section presents an overview of the NDMAS process and discusses the input and output from the process and the major components and activities of the system. This section focuses on the principle way NDMAS does business. Special cases can be accommodated to support data generators and end users.

Research activities conducted by the VHTR Program result in data records. These data records are provided to NDMAS through the use of drop boxes or email messages. The supplied data files are stored in native format in the archive. The raw data to be used in analyses and displayed on the Web are captured, transformed, and stored in a structured query language (SQL) database. Views are constructed from the SQL tables to associate data with metadata to give context to the measurements. A data mart is maintained where data extracted from the vault are transformed into formats to support Web-based reports. NDMAS maintains a secure Web page that contains reports, graphs, and analyses. End users can access and download the data through the Web page. The NDMAS process is summarized in Figure 1, which shows the primary components of the system and the general flow of data from the data generator to the end user.

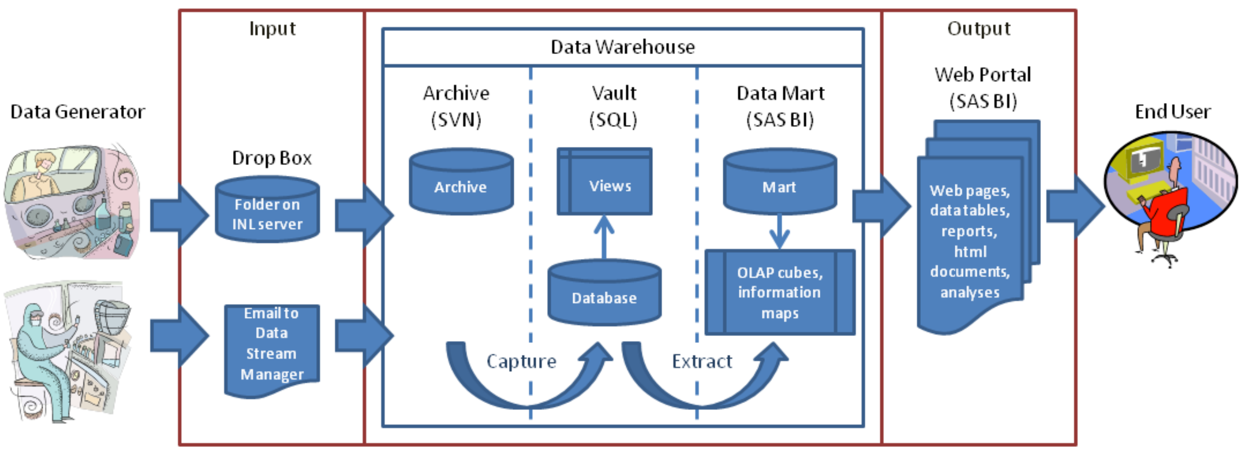


Figure 1. NDMAS data processing model.

2.1 Input

Inputs to NDMAS consist of scientific and engineering data collected by the VHTR Program. These data are transferred to NDMAS in a many ways. The files can be delivered manually, stored on a server file, or automatically stored in a file on the server. Raw data files can include PDF files, delimited text files, Excel spreadsheets, and binary files. Inputs can even consist of paper forms or PDF files of scanned images that have to be manually typed into the computer. This section discusses the properties of the incoming data.

2.1.1 Data Type

Data collected by the VHTR Program will be used for a variety of purposes. Much of the data at the INL are collected within a Nuclear Quality Assurance (NQA) NQA-1 compliant quality assurance (QA) program. Data collected at universities or other program partners may be collected under other QA programs. The data transmitted to NDMAS must be labeled by the data generator to provide a basic understanding of the provenance of the data. A classification system has been established and as defined in MCP-2691, "Data Qualification." Four data types are identified. The applications of these data types for both the NQA-1 Program and Qualification are displayed in Table 1 and described below.

Table 1. Properties of data types.

	Data Type			
	A	*B*	*C*	*D*
NQA-1 Program	Yes	Yes	No	No
Qualification	Yes	No	Yes	No

- *Type A*. Data collected within an NQA-1 QA program that must be collected to meet specific requirements for data use. Data collectors must verify that test requirements were met. Independent verification may be used to ensure that all other requirements were met. This category includes data collected to generate the licensing data such as gamma spectra analyzed to derive isotope activity, stress-strain data analyzed to derive a strength modulus, or a photomicrograph used to measure the thickness of an outer pyrolytic carbon layer. *Type A* data are qualified as directed in MCP-2691.
- *Type B.* Data collected within an NQA-1 QA program for which specific requirements cannot be defined or for which independent verification is not needed or desired. The Advanced Gas Reactor (AGR) gross gamma data are a good example of *Type B* data. Some parameters may be monitored for information only. In some cases, it may not be possible to define the needed accuracy and precision requirements for a parameter because procedures or end use has not yet been established.
- *Type C*. Data collected by entities that do not follow an NQA-1 QA program, but which need to be qualified. For example, data gathered by foreign partners to the program may be gathered under an ISO-9001 QA system. These data could be qualified by showing that the ISO-9001 QA process was functionally equivalent to the NQA-1 process. *Type C* data are qualified as directed in MCP-2691.
- *Type D*. Data collected outside an NQA-1 QA program that are not examined but are used "as is." Examples of *type D* data are existing data from the literature or data collected by a subcontractor that does not follow an NQA-1 QA program. VHTR Program researchers want to store the data in NDMAS, perhaps to be able to compare historical data to new data collected by the program. *Type D* data are archived, captured by NDMAS, and displayed to end users "as is."

2.1.2 Data Category

During the planning and design phase of a data collection activity, the data generators and data users follow MCP-2691 to plan for data qualification and to facilitate the interface with NDMAS. NDMAS serves as a satellite file location for data collected and used by the VHTR Program. In terms of data storage, management, and access, the performing organization determines the disposition of the data based, primarily, on the need for access to the data. Data disposition falls into four categories:

- *Records Service Center*. Data records that are not likely to be needed again soon are sent to the INL Records Service Center.
- *Archive*. Native data files that may be needed as backup information, audit response, or investigation of anomalies, including the native data files from which data are captured into NDMAS.
- *Capture and Display*. Data that are captured from various native formats into consistent electronic formats and made available to program participants for viewing on the Web and downloading for analysis and interpretation.
- *Time Critical*. *Capture and Display* data that, in addition, should be displayed on the Web as soon as they have been captured, before qualification, to allow experiment progress to be tracked.

Note that the data's disposition depends on access, not qualification.

2.1.3 Data Delivery

The timing, frequency, and delivery method of data generated by the VHTR Program to NDMAS vary from data stream to data stream. Therefore, the process for the passing of data to NDMAS is adjusted for each stream as appropriate for that stream. Data are transmitted to NDMAS by e-mail, removable electronic media, paper, or by placing the data on a network-accessible server where the data can be accessed (referred to as a drop box). Examples of data package delivery options include:

- Monitoring data from irradiation experiments collected from the ATR are manually placed on a network server in the form of delimited text files at about 1-week intervals. They are uploaded from the network server to the NDMAS server when they are made available by ATR Operations.
- The ATR Reactor Data Acquisition System (RDAS) and the Capsule Distributed Control System (CDCS) automatically places files into a drop box on an INL server. An automated process periodically examines the drop box, and reads any new files. Requirements for this data transfer are documented in TFR-747, "RDAS-CDCS Data Transfer to NDMAS."
- Data from graphite characterization activities, generally consisting of one data package in the form of Excel spreadsheets per Advanced Graphite Capsule (AGC) project, are placed in a drop box on a network server. New data rarely becomes available, so all data activities are manually performed.
- Gross gamma data are received daily in binary files placed in a drop box on an INL server.

Data submitted to NDMAS for archive or display must be accompanied by INL Form 435.78 as described in PLN-3319, "Records Management Plan for the VHTR Technology Development Office Program." This form can accompany a single data package, or may cover an entire data stream under certain circumstances. Data streams that consist of many identical data packages submitted routinely over time, such as irradiation monitoring data, ATR operating conditions, and gross gamma monitoring, can be covered by one form for an entire project. For data streams that consist of a few data packages distributed over time, such as graphite characterization, fuel fabrication, or PIE, each data package is commonly accompanied by a Form 435.78. Regardless of the design, each file submitted to NDMAS must be associated with a Form 435.78.

2.1.4 File Types

Data streams generated by experiments come in a wide variety of formats. Data in a machine-readable format can be captured directly from the native file, while data not in a machine-readable format requires manual entry. Examples of data package formats received include:

- *Delimited text files*. This is the preferred method of data delivery. Some delimited text files are well behaved with simple column headings in the first row and data lined up below in the subsequent rows. Some of the text file deliverables inherit their format from older computer printouts, and contain many blank rows, repeating headers, and extraneous information. All can be read relatively easily with text parsing software.
- *Excel*. Excel is a common file format for data. Much of the data that has been transferred to NDMAS has been in the form of Excel files. SAS® has good tools for reading older xls Excel files, but limited ability to parse newer xlsx files. One problem with Excel files is that data generators attempt to make them attractive by merging cells, inserting multiple rows of headings, and leaving blank columns and rows in the data. For some data streams, it has been necessary to reformat the Excel files so they can be read. Office 2007 xlsx files can be opened in Excel and saved as an older format xls file.
- *Portable Document Format (PDF)*. The fuel fabrication data usually comes in the form of PDF documents generated from scanned images, not converted from the original doc or xls file. As a

result, the data are images. For PDF files like this, the data are manually typed into spreadsheets to be captured into NDMAS.

- *Binary*. Binary files can be read as long as the encoding of the data is known. Gross gamma data are delivered to NDMAS in the form of binary files. The files are read and converted into SAS data sets for processing.
- *Database*. No data have been transmitted to NDMAS in the form of database tables. Such data could be readily captured by creating an open database connectivity interface to the source database.

2.1.5 Documentation

NDMAS does not maintain copies of documents. INL EDMS maintains electronic copies of documents. Documents provide metadata about the data in NDMAS, and NDMAS maintains a list of document references for the data. Tables in NDMAS hold information on references and are linked by foreign keys to data streams. The references that support the data in a data stream can be found using the foreign keys. An EDMS link field is included with the reference when available. However, EDMS documents can only be accessed from inside the INL firewall.

Documentation that NDMAS should receive for data streams includes:

- Planning documents for data collection
- QA documents for data collection
- Publications and reports that present the data results
- Documentation of data qualification activities and conclusions.

NDMAS can store a copy of documents that have been approved for external release on the external server and provide a link to the document for end user access from outside the INL firewall.

2.2 Data Warehouse

The data warehouse is where the VHTR Program data are stored by NDMAS. The warehouse is divided into three areas as shown in Figure 1 (above). The difference is in the amount of processing that has been performed on the data and the suitability of the data for use. The three areas of the warehouse are the archive, vault, and mart. The archive holds original data files in native format. These data can be difficult for analysis programs to access. The second area is the vault. Data in the vault are in an SQL database in a standard format. The third area of the warehouse is the mart. In the mart, data are in the form of SAS datasets and have been formatted for access by SAS EBI Web tools, and data are associated with metadata to provide provenance and context to the data.

2.2.1 Archive

The archive is the repository for VHTR Program native data files. The primary purpose of the archive is to store the original data files delivered by the data generators that contain the data that are captured by the NDMAS system. The files serve as long-term storage to allow verification of the capture process. Other VHTR Program data files can be stored in the archive if the data generator determines that the archive provides a better means of access to the data than placing the data files in EDMS through the Record Service Center.

NDMAS maintains the archive on the NDMAS server in a directory named NGNP_DATA. Below that level, the archive is organized by the standard NDMAS structure based on projects and data streams as discussed in Section 5.1.

The archive retains the files as delivered from the data generator. At times, changes must be made to the original data files. For example, the files as delivered might not be efficiently read for capturing, or errors are found in the original files such as mistyped specimen identifiers. Before changes are made to a file, the file is added to a subversion system so that all changes are documented as to who made the changes, what changes were made, and the justification for making the changes.

2.2.1.1 Prerequisites

Prerequisites for storing data in the archive are:

- Form 435.78 accompanies the data (see Section 2.1.3)
- Access is needed to data in files that cannot be attained through EDMS.

While the primary purpose of NDMAS is to store VHTR record data, data that are not a VHTR Program record can also be stored in NDMAS at the request of a data stream owner. Such data might be historical data, or data that provide context to the VHTR Program data.

2.2.1.2 Activities

Activities associated with Archiving data include:

- Store data
- Enter data files into a change control system
- Maintain a change control system.

2.2.1.3 Definition of Success

Success is defined as:

- Data are readily available
- Any changes are tracked
- Original data can be retrieved.

2.2.2 Vault

The vault serves two major purposes for data storage in NDMAS. First, the data in the vault are efficiently stored in an electronic format. Second, the data in the vault are in a controlled, secure environment where the quality and provenance of the data can be rigorously maintained. Almost all of the data captured by NDMAS are stored in the vault. Gross Gamma data are currently the only *Capture and Display* data not stored in the vault.

The vault consists of a Microsoft SQL Server database maintained on INL servers. The database consists of many tables designed to hold the data from the variety of data streams generated by the VHTR Program. The design of the database is documented in SDD-228, “NDMAS 2.0 Database.” The database is optimized for efficient storage of data.

Views are virtual SQL tables based on SQL statements that can combine data from multiple database tables. The vault contains many views that associate data values with context information about the value, such as where and when it was measured, methods used to take the measurement, and data quality indicators. The views provide an efficient means to access the data in the vault.

2.2.2.1 Prerequisites

Prerequisites for storing data in the vault are:

- Data capture has been verified
- Testing is complete and data state flags assigned
- Referential integrity exists to plan and event tables.

2.2.2.2 Activities

Activities associated with the vault include:

- Make corrections to data
- Track changes
- Update qualification state flags when data qualification is complete.

2.2.2.3 Definition of Success

Success is defined as:

- No undocumented or unjustified changes to data
- No loss of data
- All data with documented history and qualification state flags assigned.

2.2.3 Mart

The mart contains data optimized for use. Data are extracted from the vault, transformed to support analyses, reports, graphs, and data tables, and loaded in SAS datasets in the mart. The mart is organized based on the projects and data streams (see Section 5.1) but also allows for the storage of data to support analyses. In the mart, data from multiple data streams are associated to allow analysis and interpretation. For example, in the vault, irradiation monitoring data, fuel fabrication data, and reactor neutronics simulation results are stored in three separate tables. In the mart, these data are associated in one data set so that the effects of irradiation on fuel properties can be analyzed.

Large data sets can be cumbersome to view and interpret. Online analytical processing (OLAP) cubes presummarize data from large datasets to allow users to view summaries quickly, and then drill down into the data if more detail is needed. OLAP cubes are built from SAS data sets using SAS OLAP Cube Studio and stored in the mart.

SAS EBI provides the ability to create information maps that describe data in OLAP cubes and data sets. The information map provides formatting, summarization, and data labeling. Information maps are built for OLAP cubes and data sets using SAS Information Map Studio and stored in the mart.

2.2.3.1 Prerequisites

The Capture and Display data category is in the mart.

2.2.3.2 Activities

Data sets and cubes are updated in the mart with new data as it is added to the vault.

2.2.3.3 Definition of Success

Marts designed for efficient access to data are formatted for use.

2.3 Extract, Transform, and Load

Data delivered to NDMAS comes in a wide range of formats that change over time for data streams that represent long-term monitoring of irradiation projects. The data are not in a suitable format for analysis. For example, measurements of graphite specimen characteristics arrive as 21 spreadsheets, each of which has similar, but not identical formats. As delivered, questions about how the properties of one grade differ from those of another grade cannot be answered. Initially, shim position data from the ATR were delivered as six-character strings of ‘I’s and ‘O’s, then delivered as six-character strings of ‘1’s and ‘0’s. Now, the shim position data are delivered as six separate variables, each taking a value of ‘1’ or ‘0’. As delivered, it is impossible to analyze the changes in shim position over time.

This section briefly describes how NDMAS takes the data from the raw data files provided by the data generators and turns it into information on Web pages. Extract, transform, and load (ETL) is data integration terminology for the process of taking data from raw files, in a variety of formats, and turning it into data that can be accessed by analytical tools, and storing it in a place that can be accessed by analysts. This occurs in two steps in NDMAS. The first ETL step is the capture of values from raw data files and loading it into the SQL database. The second such step is extracting data from the SQL database and loading it into the data mart. A detailed description of these two steps is provided in the next two Sections.

2.3.1 Capture

Raw data files are generally not in a format that the data can be readily accessed for display and analysis. Data may be spread over multiple files, or data within a file may be in a format that is difficult to read. Data in the *Capture and Display* disposition category are captured from raw data files and transformed into formats that can be pushed to the SQL database. Data are captured using codes that have been tested, reviewed, and are maintained under configuration control. Captured data to be stored in the SQL database are assigned necessary foreign key ID values from tables in the database, and are then pushed to the SQL database. Captured data are verified as being equivalent to the raw data. Data may also be screened for anomalies that might indicate data are not an accurate representation of the system or object being measured. Screening for anomalies is implemented on a data-stream by data-stream basis, depending on the needs and conditions of the individual data streams.

2.3.1.1 Prerequisites

The prerequisites for capturing data are as follows:

- Raw data file stored in the archive
- Codes used to capture data tested, reviewed, and in code repository
- Data type (A, B, C, D) identified by data generator
- *Capture and Display* data category.

2.3.1.2 Extract

SAS Enterprise Guide (EG) programs, or other programs capable of parsing incoming data streams, are used to read data from raw data files. SAS has functions to read both delimited text files and Excel spreadsheets. SAS can also read binary files if the file structure is known. The EG programs are used to create SAS datasets of the incoming data. The SAS datasets are then transformed using SAS programs and functions to prepare for loading into the SQL database.

2.3.1.3 Transform

Data are transformed so that they are compatible with the NDMAS SQL database or other storage media. For example:

- Take various date formats (Microsoft, UNIX, text strings) and turn into SQL standard date/time formats. Remove any daylight savings time offsets and put time in Mountain Standard Time (MST). For variables with a time-support component, adjust the time to the start of the interval. Add the time-support interval to the record.
- Convert units to standard set of units.
- Calculate averages or other statistics from individual measurements.
- Assign foreign key information to give context to the data.
- Separate and format the data to align with the table schema in the database. This commonly involves splitting the incoming data into multiple SAS datasets according to measurement variable (gas flow, gas pressure, temperature, etc.) and transposing the data so that there is one column of values with multiple rows for the multiple measurement locations.

2.3.1.4 Load

Once the data are transformed to the format needed for the SQL database, the data are loaded into the NDMAS database. There are two options for load, either load the data directly into the vault, or add the data into an intermediate staging database. The staging database contains copies of the data table structures from the vault and provides a location to perform accuracy testing on data prior to loading it into the vault. Once data in the staging database are verified, they are pushed to the vault. Verification activities depend on the data stream.

2.3.1.5 Definition of Success

Successful completion of the capture step results in:

- Data pushed to vault are equivalent to data in raw files
- Anomalous or failed data identified with data state flags
- Qualification state flags are assigned to the data
- Data in the vault are joined to tables containing metadata and context information by foreign keys.

2.3.2 Deploy

Extracting data from the vault and loading it into the mart is easier than capturing the data from raw files because the initial format is well defined. The data still must be transformed to support data analysis and Web pages and be loaded into the mart.

2.3.2.1 Prerequisites

Prerequisites for this step are as follows:

- Data must be capture verified
- Data must exist in the vault or other controlled storage location (such as for gross gamma data).

2.3.2.2 Extract

Views in the vault associate data with metadata and context information such as location, instrument, measurement units, and data stream information. EG projects are used to extract data from the vault and save the data in temporary SAS datasets for transformation. SAS can directly access views and tables in the database as if they were SAS datasets.

2.3.2.3 Transform

SAS EG projects are used to transform data as extracted from the vault into formats that support Web reports, data analysis, and graphics. Examples of transformations include:

- Transpose data from separate single columns into multiple columns by component
- Join columns from different tables on date/time and location to associate multiple variables collected at the same time and place
- Build OLAP Cubes
- Create information maps from cubes and data sets.

2.3.2.4 Load

Save data sets, OLAP cubes, and information maps in the data mart.

2.3.2.5 Definition of Success

Data in the mart are clearly associated with metadata and context information (response variable, location, project, etc).

2.4 Analysis

NDMAS can be used to explore relationships in the VHTR Program research data at a number of levels. At the simplest level, researchers can view graphs of data and think about possible relationships that might exist between various physical quantities in the experiment.

Intuitive models that come from data visualization can be checked by empirical studies. One can estimate parameters, such as averages within groupings of data, and observe which groupings have better outcomes. Formal statistical tests can be performed to identify differences that are beyond the realm of normal variation. At a higher level, one can formulate mathematical relationships that might exist between various attributes or measured quantities and fit these to the empirical data to see what relationships best represent the data.

More involved data analyses go beyond just empirical patterns, they may include known physical relationships between the quantities under study. Physical and empirical models can be combined to even better characterize relationships among the measured responses.

A final level of modeling involves simulation of dynamic conditions as they propagate in a complicated system over a period of time, such as thermal simulations of fuel temperatures. Such models are not part of the NDMAS modeling capabilities. NDMAS may support the development of such models, provide data as inputs to the models, and store the outputs. Some of the experiments planned for the VHTR Program will produce data to calibrate and validate such models, and NDMAS analysis capabilities will facilitate the study of those data as well as other data generated by the VHTR Program.

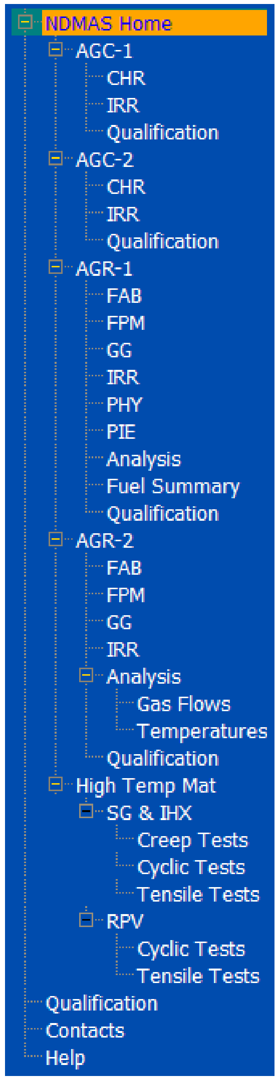


Figure 2. Snapshot of current Web page navigation hierarchy.

NDMAS supports analyses of relationships between quantities in particular data packages. It also facilitates studies of data across data packages, such as from one reactor cycle to another. One can also combine data and study relationships among quantities from more than one data stream. ATR operating conditions data, for example, are combined with VHTR Program irradiation experiment data for routine plotting and for analyses of controls on temperature. Data from different components, such as different fuel capsules, can also be visually compared using NDMAS. Particular modeling efforts can be geared to the objectives of each VHTR Program experiment, or to resolve particular questions posed by other VHTR Program researchers or stakeholders.

A complete description of NDMAS analysis capabilities is provided in *NGNP Data Management and Analysis System Analysis and Web Delivery Capabilities* (Gentillon et al. 2011) so will not be discussed further here.

2.5 Output

The real power of NDMAS is the ability to provide access to data for end users through a Web browser. End users can access data from any computer connected to the internet from anywhere in the world. This facilitates collaboration among the many VHTR Program partner organizations, and allows international partners, such as France and South Africa, to participate in data analysis directly. NDMAS uses a SAS Enterprise Business Intelligence (EBI) Web portal to make the experimental data and analytical results available to the VHTR Program research community.

A complete description of NDMAS data delivery capabilities is provided in *NGNP Data Management and Analysis System Analysis and Web Delivery Capabilities* (Gentillon et al. 2011).

2.5.1 Web Delivery Portal

NDMAS uses the SAS EBI platform to deliver experimental results from VHTR Program data generators to the community of interested stakeholders over the Internet. The results are delivered in two primary forms: downloadable data tables, and graphs. However, many details and subtleties are considered in the process of developing Web page content. These considerations include the nature of the data being displayed, the types of tabular data that would be useful, the grouping of the data so that vast amounts of data can be shown in an organized presentation, and the types of graphs or other displays that would facilitate display and understanding of the data. The hierarchical organization of the Web page is shown in Figure 2. The organization follows the general NDMAS organization structure discussed in Section 5.1.

2.5.1.1 Prerequisites

Prior to publishing a Web page, the following prerequisites must be met.

- Approved change request for codes and Web page(s)
- STIMS approval
- Security in place to limit access to data.

2.6 Special Cases

Not all of the data streams are handled the same in any system. This section discusses data streams that are managed differently because of special characteristics or special circumstances.

2.6.1 Gross Gamma

Gross gamma data consists of nominal 8-hour sequential gamma count data in a binary file format from each of the capsule fission product monitoring systems (FPMS). These data are processed by NDMAS to provide daily graphical Web displays for VHTR project leads to aid in their understanding of general experimental conditions. The data are archived and processed for graphical Web displays by NDMAS but are not stored in the vault or qualified.

The binary files are generated by the FPMS gross gamma detectors on a periodic basis and then copied by FPMS staff to an NDMAS share location on the isasapp server (isasapp\ngnp_agr2_fpms\Spectra\). Twice a week, NDMAS processes these data using SAS code, which converts the data into ASCII text, combines the data into single detector/capsule-specific SAS data sets for plotting, stores the processed data sets in an NGNP_Data archive location, then generates plots of the data for Web reports using SAS plotting routines and stored processes. These plots are displayed both on experiment (e.g., AGR-2) home pages and on a designated Gross Gamma (GG) Web page. Plots are generated for: (1) the most recent 7 days of data, (2) the most recent 24 hours of data, (3) any user-specified date/time interval, and (4) statistical summaries (mean, standard deviation, maximum) of the experiment to date.

3. Logical Data Model

The data model is a schema for the representation of point measurements using tables linked by associations or relationships between key fields. The model is implemented in a relational database schema to facilitate integrated analysis of large datasets collected by multiple investigators. Observations are stored with sufficient metadata about when, where, and how the data were collected to provide context for the values. Metadata associated with data quality are also included in the schema to provide provenance for the data. The supporting information will allow the data to be unambiguously interpreted and provide traceable heritage from raw measurements to useable information. The design is based on exposing each observation, or a small subset of closely related observations, as a record. This allows associating metadata, qualification states, and context information, as directly as possible with the individual (or closely associated small group of) measurements.

3.1 Basic data concepts

3.1.1 Data Value

Each measurement of a property on an object or process needs to be located in time, space, and identity as shown in Figure 3. What is being measured must be uniquely identified so that the end user knows what response variable was measured. This includes information on the units of measure and if the value is a point sample, physical composite over time or space, or a statistical summary of measurements. Good identification of what is being measured is needed to determine if one value is comparable to another value. The measurement must also be located in time (vertical axis shown in Figure 3). This may be time, for a time series, but could also be a relative position is a sequence of events. For example, graphite specimens are characterized, then irradiated in ATR, and characterized again after irradiation. Measurements of properties must be accurately identified for position in this sequence as before, during, or after irradiation. Measurements could also be taken over a range of temperatures, and the temperature would indicate the position along the sequence rather than time. Finally, the location at which the measurement was collected must be known. Location can also include the identity of a specimen subject to measurement. Location can be hierarchical, in that location may require knowing both the specimen number and the specific location on the specimen where a diameter was measured. The database must provide this contextual information to allow unequivocal interpretation of the measured values.

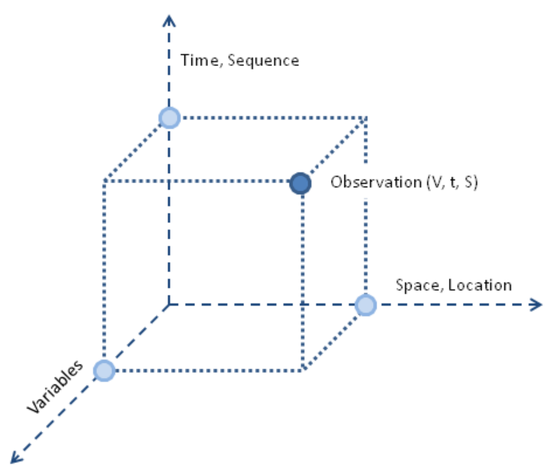


Figure 3. Observation values are located by the identity of the measured variable (what), the position in space (where: location, object being measured), and time (when: time or position in a processing sequence).

Metadata are data about the data. Metadata provide context for the data and are critical to evaluating the quality of the data. Metadata would describe how the sample was collected, how the measurement was made, conditions the specimen was exposed to during irradiation, and many other pieces of information. The structure of metadata is highly variable among the VHTR Program data streams and poses the biggest challenge to simplifying the structure of the data model. Each stream has specialized metadata, which leads to developing different table structures for different streams.

The measurement value, or a small set of closely related measurement values, is the focus of the data management structure. A single number, or a small group of closely related numbers, form a single record. This permits tracking metadata associated with each value, rather than attempting to generalize the metadata. Data quality indicators are also tracked at the record level providing provenance and traceability of the individual, or a small number of closely related, measurements.

At this time, NDMAS does not have capabilities to handle multidimensional arrays. Multidimensional arrays are datasets where the set as a whole is interpreted rather than as a collection of individual numbers. Techniques such as network common data form (netCDF) exist to manage such data. Such techniques can be adopted when array-based datasets are provided to NDMAS for storage.

3.1.2 Organizational Hierarchy

The VHTR Program is divided into several projects. Each project consists of a series of experiments. Within an experiment, the incoming data are organized into data streams. Data streams are subdivided into data packages. The design of the data storage structure is based on a hierarchy of:

- Project
  - Experiment
    - Data stream
      - Data package
        - Data value

The VHTR Program is divided into three currently active projects: AGR, AGC, and High Temperature Materials (HTM). Each project is divided into a series of experiments (such as AGR-1, AGR-2, AGR-3/4, etc.). For NDMAS, an experiment is defined as a specific task identified in the program work breakdown structure as defined in PLN-2494, “VHTR Technology Development Office Program Management Plan” for AGR and AGC projects and PLN-2803, “Next Generation Nuclear Plant Reactor Pressure Vessel Materials Research and Development Plan,” for HTM. One or more data streams will be generated for each project as shown in Table 2. A data stream is a particular work flow pathway along which related data flow into NDMAS. Data are received in batches, called data packages. The number of data packages from an experiment ranges from one to dozens, depending on the data stream. Data packages are associated with a specific data stream and are aligned with some logical pattern or periodicity with which data are received into the system. For example, graphite characterization data for the AGC-1 preirradiation data stream were received in one data package, reactor operations and capsule conditions monitoring data have been received in biweekly data packages for the duration of the irradiation experiment, which may last up to 36 months, and portions of the fission product monitoring data are received at the end of each reactor cycle. The data element is the single variable value recorded that provides information about the system or object being measured. Data elements are divided into response elements and attribute elements. In general, response elements are numeric values that describe the response of the object or system. Examples include pressure, temperature, and elastic modulus. Attribute elements generally describe the object or system being measured, or provide categorical or spatial information about the object such as thermocouple composition, graphite grade, or capsule position.

The data structure of a data stream for a project is generally very similar across experiments. Therefore, one design concept for the database is to build tables for data streams within a project across multiple experiments. So far, this has worked to hold all the data for the experiments.

Table 2. Data streams associated with the three VHTR Program projects currently being conducted.

	Projects		
Data Stream	AGR	AGC	HTM
Material property characterization	X	X	X
Irradiation monitoring	X	X	—
Fission product monitoring and gross gamma	X	—	—
ATR Operating Conditions	X	X	—
Post irradiation examination	X	X	—
Neutronics and thermal simulation	X	X	—

3.1.3 Dates and Times

Unambiguous interpretation of date and time requires specification of the time zone. Because most of the time series data are collected at INL, NDMAS stores time in Mountain Standard Time. Any daylight savings time shifts in the incoming data are removed.

A set of time-series data are characterized by three factors: extent, spacing, and support as shown in Figure 4. Extent is the duration over which the measurements were collected. Extent is defined by the minimum and maximum times for the series. NDMAS stores this information (when applicable) with each experiment in the Project table.

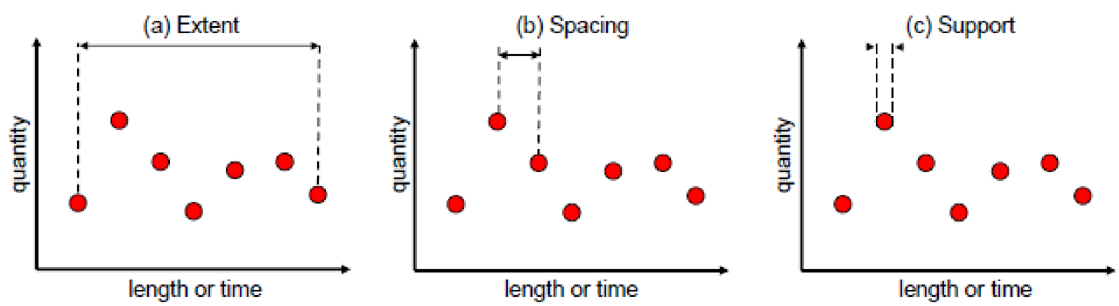


Figure 4. Time series data are characterized by (a) extent, (b) spacing, and (c) support.

Most of the ATR operating conditions data and capsule monitoring data are collected at short, regular, periodic intervals. This spacing ranges from 1 hour for data collected early in the VHTR Program to a spacing of 5 minutes currently for CDCS data and 1 minute for RDAS data. NDMAS does not store information on spacing. If of interest, it can be calculated from the date time values stored with the data.

Time support is a time interval over which data are collected, and then processed to produce a single reported value that represents the support interval. For example, temperature could be measured over an hour and then reported as a single average value for that hour. Power could be measured over an hour and then reported as the cumulative power generated over that hour. Support and spacing do not have to coincide, but support should be less than or equal to spacing. Prior to summer of 2012, data collected by the CDCS for irradiation experiments at ATR were reported as average values over a support time interval. The support interval and the spacing interval coincide for CDCS data. This time support interval has varied over the duration of the VHTR Program between 1 minute and 1 hour. Therefore, the database stores information on the time support and the statistic calculated for the interval. In the summer of 2012, when the automated CDCS feed was initiated, the CDCS data became single measurements, and time support is considered instantaneous for CDCS data after that time. For NDMAS, data collected as a summary statistic over a time interval are reported at the start of the time interval.

3.2 Time Series Data

A generalized database schema was developed to hold time series data collected during irradiation experiments. This schema is used for ATR operating conditions, AGR and AGC irradiation monitoring data streams, fission product monitoring data stream, and neutronics and thermal simulation data streams. The schema includes a hierarchical design for specifying the location of the observation and a table to hold the events associated with the time series. The hierarchical location design allows the addition of as many columns as necessary to store location-specific metadata. The events are date-time values, but may also include other time-related values such as effective full-power days or ATR cycle. The general schema is shown in Figure 5.

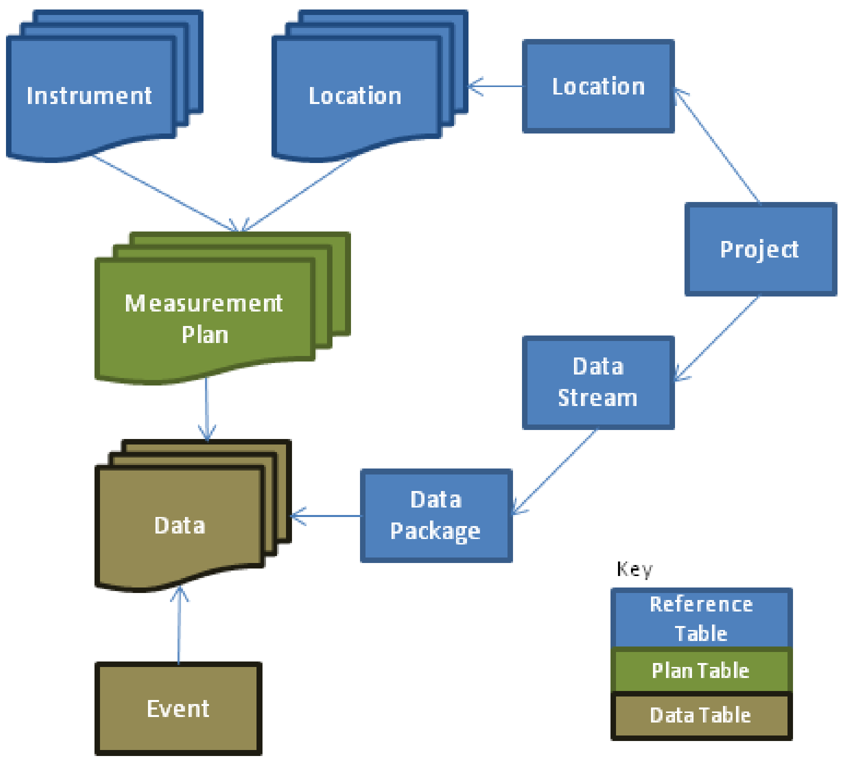


Figure 5. General data schema for time series data adopted for the NDMAS database design.

3.2.1 Measurement Location

Measurement location is hierarchical starting with the experiment. Each experimental apparatus placed in the reactor consists of one or more capsules. Each capsule has specimens (either fuel or graphite) stacked within the capsule in multiple channels (seven for AGC, three for AGR). The position of each specimen is important for linking the specimen to the temperature and neutron flux the specimen is exposed to. Table 3 presents an example of how the design has to support the linking of data from a wide range of data streams with very different characteristics.

Table 3. Example of how the design has to support the linking of data from a wide range of data streams with very different characteristics.

AGC Location Hierarchy	AGR Location Hierarchy
Capsule	Capsule
Channel	Compact
Specimen	Cell
Gas Gap	Gas Line

Gases are used to control the temperature in the capsules by altering the thermal conductivity around the capsule. Gas pressure, gas composition, and gas moisture content are measured in gas lines or in gas gaps in the AGR and AGC capsules, respectively. The gas gap or line is a subcomponent of the capsule, and so occurs at the same level in the hierarchy as the channel or compact.

Location tables contain metadata about the location. At the moment, there is little information in the database about the locations. It would be easy, for example, to include information about fuel for each capsule such as fuel type and owner.

3.2.2 Measurement Instrument

To collect data at a location, an instrument is installed that is sensitive to the parameter being measured. NDMAS is currently not storing specific information about instruments with the exception of thermocouples. Therefore, in the instrument tables, there is one generic instrument that measures a specific parameter. For moisture and pressure sensors, there is only one sensor. For gas composition, there is one sensor for each type of gas used in the experiments. The design allows a record for each instrument installed in the experiments if this information is needed in the future. Instruments installed in the AGC and AGR experiments include thermocouples, pressure transducers, moisture sensors, gas flow meters, load cells, and position sensors.

3.2.3 Measurement Plan

An instrument is installed at a location to collect measurements. There may be metadata about the installation, for example the position of a thermocouple in a capsule. Therefore, plan tables contain foreign keys to the location and instrument tables. The pair of foreign keys must be unique. The plan table also contains information about the time support for measurements made at a location. Also included in this plan is the information that links this measurement to the incoming data stream. The RDAS and CDCS feeds from ATR are based on keywords to identify the variable and location. These key words are included in the plan table, so that by making a join between the incoming data and the plan table, the plan table ID can be associated with the incoming data value.

3.2.4 Measurement Event

For time series data, the date and time the measurement is recorded must be associated with the measurement. The NDMAS database design uses separate event tables to record the measurement events and links the events to the values using foreign keys in the data tables. As a minimum, there are two data fields in the event tables. The first is the source date-time, as provided by the data generator. The source date-time is adjusted to Mountain Standard Time, as necessary, and adjusted to the start of the time support interval, as necessary. The adjusted time is held in the standard date-time field. Other time-related data fields may be included in the event table, depending on the data stream. Some additional columns include effective full-power days and ATR cycle.

3.2.5 Measurement Data

Measured values are stored in data tables that are linked to plan tables and event tables using foreign keys. Referential integrity is enforced. Data tables consist of one or a few columns of values. Each column can contain only one type of data. If there are multiple columns of values, they must link to the same plan table and event table entry. Therefore, most data tables have a single column of values. Separate data tables are used for different types of data; gas flows, gas pressure, and gas moisture are three different tables. Each record in the data tables also has columns to track data quality flags.

3.3 Specimen Data

Fuel fabrication, PIE, and graphite specimen characterization are examples of measurements of properties on specimens. This type of data is managed in tables with different structures than the time series data. The structure of specimen data also varies considerably more than the time series data, so database schemas for the different specimen-oriented data streams vary considerably to address the different characteristics as shown in Figure 6. This section provides a high-level conceptual model for specimen data. Details of the database design are presented in the SDD-228, “NDMAS 2.0 Database,” design document. The following sections discuss the types of tables used in the specimen data database schema.

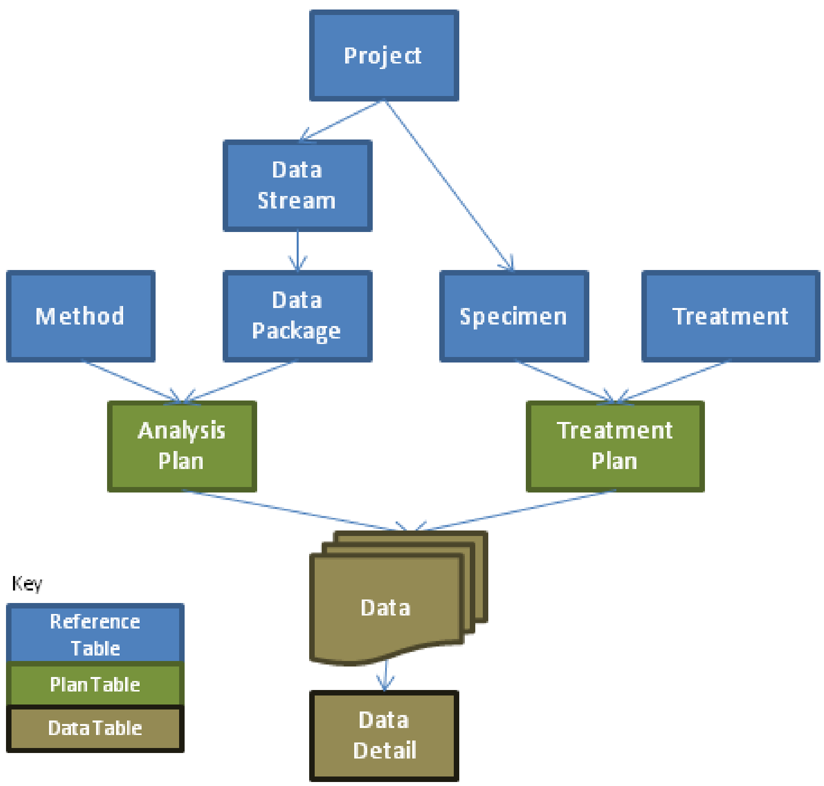


Figure 6. General data schema for object characterization data adopted for the NDMAS database design.

3.3.1 Measurement Object

The measurement object tables contain the list of objects that have been characterized. This would be a list of graphite specimens or a list of fuel compacts. Each type of object has its own table, allowing the columns in the table to be customized to the particular type of object.

The position in the graphite billet from which the specimen was cut is important, so there are a series of fields that describe the location in the billet. There are also fields for specimen type and orientation of the specimen with respect to the grain orientation. Such information is not relevant to fuel compacts. Fuel compact object tables contain information on particle layers and layer composition.

3.3.2 Measurement Method

Very important metadata for future data users is the method used to collect data on the objects. This is needed to understand whether data from different sources are comparable. The measurement method table lists analytical methods.

3.3.3 Treatment

Objects are often processed through a series of steps, with measurements taken before and after processing. In the data table, the same parameter is measured on the same object, so the treatment of the object at the time of measurement is necessary to differentiate the two measurements. The most common treatment sequence for the VHTR Program is preirradiation characterization followed by irradiation and PIE. This table holds a list of treatments starting with ‘None.’ Treatments are data stream specific.

3.3.4 Analysis Plan

The analysis plan table joins the data package with the analytical methods used to collect the data in that data package. This allows for changes and updates in analytical methods over time. There will be a record of whether a particular sample was analyzed with the 2008 or 2012 version of an analytical method. Usually, this will result in one record in the plan table because typically only one method is used for data in a single data package. This is not always the case, but the use of the plan table this way also allows for multiple analytical methods to be applied to a data package. Individual measurements in the data table would point to the correct plan record.

3.3.5 Treatment Plan

The treatment plan is the join of the specimen and the treatment. This creates a unique combination of specimen and treatment to differentiate multiple measurements of the same variable on the same specimen though multiple steps of a process.

3.3.6 Measurement Data

Measurement data tables contain the measured values. Records in measurement data tables contain a few related columns. Because the data table is linked back to the Measurement Plan table, all the values in a single record must be measured by the same method. This is the basis for defining the columns that can be placed into a measurement data table. Each row has foreign keys to the measurement plan table and the treatment plan table. Referential integrity is enforced. Each row also contains data quality flags.

3.3.7 Measurement Data Detail

More detailed data are collected for some measurements. This table provides for multiple measurements for the same measurement plan and treatment plan ID values. As an example, measurements of thermal expansion are taken over a series of temperatures from 100 to 800°C. This table contains the detailed measurements at the various temperatures. Fuel compact diameters are measured at multiple locations along a compact. The detail table holds the individual measurements. If a mean is calculated, it would be stored in the data table.

4. ROLES

This section discusses the roles people take on in interacting with the NDMAS system. The responsibilities are designated to the roles. Individuals assigned to these roles can delegate the performance of the duties.

4.1 Data Generator

The data generator is the organization or individual(s) that collects the data. For the VHTR Program, the generator is generally taking the measurements themselves. However, the generator may subcontract the data collection to other organizations, or may take data from the open literature. It is the generator that transmits the data to NDMAS.

4.1.1 Data Stream Owner

The data stream owner is the line manager with programmatic responsibility for collecting the data from a data stream.

4.1.2 Responsibilities

The originator is responsible for completing INL Form 435.78, "VHTR Technology Development Office Information Input Sheet" and submitting it to NDMAS along with the data. NDMAS forwards the form to the VHTR Records Service Center for processing. The form is a requirement of the VHTR Program records management plan (PLN-3319) for submitting data to a satellite file location.

The originator is responsible for the quality of the incoming data and for documenting the data quality. NDMAS is responsible for identifying data in the database with the provided data quality indicators and maintaining the integrity of the association. To communicate data quality, the data generator labels incoming data as *Type A, B, C*, or *D* (see Section 2.1.1 for an explanation of data types). The originator is responsible for qualifying *Type A* and *Type C* data; NDMAS can assist in data qualification. The originator also determines the disposition of the data, such as whether the data are to be kept in the archive for future reference or captured and displayed on the Web. The originator also determines if the data are to be captured and displayed immediately (time critical data) or after the data have been qualified.

As the technical experts on the data, the originator is responsible for making decisions about what variables are to be displayed and distributed via the Web. Where there are access limitations on the data, the data generator is responsible for identifying those limitations to NDMAS and to define individuals or groups who can access data.

4.2 NDMAS Team

The NDMAS team operates NDMAS. This section describes the responsibilities of the NDMAS team. Team functions can be delegated, but not the responsibility.

4.2.1 Data Stream Manager

The data stream manager is responsible for:

- Acting as the interface between NDMAS and the data generator and working with the generators to determine what data will be stored and how it will be displayed on the Web

- Helping the generator understand the concepts of records management and data qualification, and assisting in qualifying the data if desired
- All NDMAS functions associated with a data stream, including:
  - Archiving data
  - Capturing, transforming, and loading data into the vault
  - Extracting data into the mart
  - Building Web pages to display the data
  - Keeping Web pages up to date if new data are received
- Making sure codes needed to capture, extract, transform, and display data are written, tested, and reviewed; ensuring the code is verified once it is in the code repository
- Ensuring any security or access limitations on the data communicated by the data generator are implemented
- Working with the data warehouse manager to develop new tables to support newly established data stream that are compatible with the type and structure of the data being received.

4.2.2 Data Warehouse Manager

The data warehouse manager is responsible for:

- Maintaining and operating the archive and vault
- Collaborating with the data stream manager to design schemas to store data in the vault
- Selecting and managing software to operate the archive and vault
- Writing and storing procedures to automate the process of testing and managing data.

4.2.3 System Administrator

The NDMAS system administrator is responsible for:

- System operations, including hardware and software, with the exception of those items assigned to the data warehouse manager
- Maintaining the SAS metadata repository, the servers on which the SAS team foundation server software operates, and the software licenses and installations
- Managing the code repository, the Team Foundation Server (TFS) change tracking system, and the SharePoint collaboration environment
- Maintaining the external Web pages and updating contents of the Web pages with new data and pages developed by the data stream managers
- Implementing and managing security and access to the system.

4.2.4 Developer

The developer is responsible for:

- Writing the code that captures data from raw files, transforms it, loads the data into the database, extracts the data from the vault, and creates data sets and graphics for Web pages.
- Creating Web pages to display data to end users.

- Following the change control process discussed in Section 5.2.

4.2.5 NDMAS Quality Assurance

NDMAS QA is responsible for:

- Maintaining NDMAS software QA documents and procedures, monitoring the change control process, and creating reports summarizing the status of work items in the process
- Reviewing codes and Web pages for conformance to requirements, moving approved codes from development to the code repository, auditing NDMAS activities for conformance to requirements, and making recommendations for process improvements.

4.2.6 Analyst

NDMAS analysts are responsible for:

- Interpreting data in NDMAS and providing reports to end users to support decision making; the scope of analyses depends on end user needs.
- Following the change control process discussed in Section 5.2 for codes used to perform analyses.

4.3 End User

End users access the NDMAS system to obtain data and analyses in the direct execution of their job duties. End users for the NDMAS system cover a spectrum of organizations, including INL, other national laboratories, universities, consultants, and national research organizations in other countries. The roles end users take include: researcher, modeler, industry, and manager. Researchers and modelers access data as input to advanced analyses and simulations. Industry will access data to support reactor design. Managers access data to support decisions.

4.4 Stakeholder

Stakeholders have an interest in the data and analyses in the NDMAS system as it affects achieving their organizational goals, but not in the direct execution of their job duties. The Department of Energy is a stakeholder in their role of encouraging development of nuclear energy and protecting their investment in materials research. The Nuclear Regulatory Commission has an interest in the data quality being clearly identified and the provenance protected to support licensing activities in the future.

5. INTERNAL PROCESSES

In order to conduct business within the VHTR Technology Development Office (TDO) QA plan, NDMAS has internal processes that have been reviewed and vetted as complying with the requirements of the QA plan. This section describes the internal processes NDMAS uses to accomplish the mission.

5.1 Organization System

The hierarchy of the VHTR Program is introduced in Section 3.1.2. This hierarchy is used as the basis for organizing things in NDMAS. Things that are organized this way are libraries in the SAS metadata repository, directories in the archive, Web pages, requirements, and change requests in the change request tracking system, and code repositories. Even the structure of the vault database reflects this organization.

The top level of the hierarchy is the experiment. In the early stages of the VHTR Program, the experiment level at the top does not create too many bins at the first level. As the program progresses, it may be that the project level will need to be imposed on top of the experiment level. Currently, the top levels of the hierarchy are AGC-1, AGC-2, AGR-1, AGR-2, AGR-3/4, Baseline Graphite, and HTM. ATR operations data is applied to many projects, and so is carried as its own branch of the organizational hierarchy rather than replicating ATR under multiple projects. For labeling purposes, the experiment label does not include the dash. For combined experiments (AGR-3/4), usually only the first digit is used (AGR3). Directories, folders, and libraries are named (e.g., AGC-1, AGR-3). Because SAS library references are limited to eight characters, the decision was made to shorten the experiment names to four characters. Full names, including dashes, are used in reports and on Web pages.

The next level of the hierarchy is the data stream level. Data streams associated with projects are shown in Table 2. The labels that are applied to the data streams in the hierarchy have been standardized to three letters for irradiation experiments. The combination of four character experiment names and three character data stream names, connected by an underscore, yields an eight-character library reference, compatible with SAS naming conventions (AGC1_IRR, AGR3_PHY) based on the following definitions:

- FAB – Fuel Fabrication
- IRR – Irradiation Monitoring
- PIE – Post Irradiation Examination
- FPM – Fission Product Monitoring
- CHR – Preirradiation Characterization
- PHY – Neutronics and Thermal Simulations
- GG – Gross Gamma.

The data stream level for Baseline Graphite, and HTM will differ, depending on the needs of those experiments. However, for irradiation experiments, where so many different people work on the data sets, this more rigid structure seemed necessary to preserve some order in the data. Below the data stream level, each data stream manager can set up any structure needed to keep track of data, programs, Web pages, etc.

5.2 Change Process for Code and Web Page Development

All software (code and Web pages) must trace to a requirement. Implementing a new requirement or changing how an existing requirement is implemented is documented using the change request work item of Microsoft® Visual Studio® Team Web Access. The change request work item is intended to walk through the needed software QA steps. When an NDMAS team member is assigned responsibility for an action under the change request process, they go to the tab related to the current stage, complete the activity, and record information in the fields provided. They then change the stage field to the next step and reassign the change request. Table 4 lists the steps and responsibilities in the change request process, which is illustrated in Figure 7.

Table 4. Steps and responsibilities for performing the change request process.

Responsible Party	Action
Developer	Identify existing relevant Requirement work item or create new Requirement work item. Enter Requirement(s).
Developer	Create Change Request (CR) work item. Enter description. Link to Requirement.
Independent anyone	Complete Requirement and change request work item reviews.
Developer	Develop code or Web page. Use subversion to track changes, provide backup, and facilitate sharing during development.
Developer	Enter software Design/Development information.
Other Developer	Conduct independent software Acceptance Test and enter test information.
Independent anyone	Complete independent software Product Review.
System Admin/SQA	Complete change request Admin Review
System Admin/SQA	Move code from development repository to production code repository
Developer	Complete software Final User Check

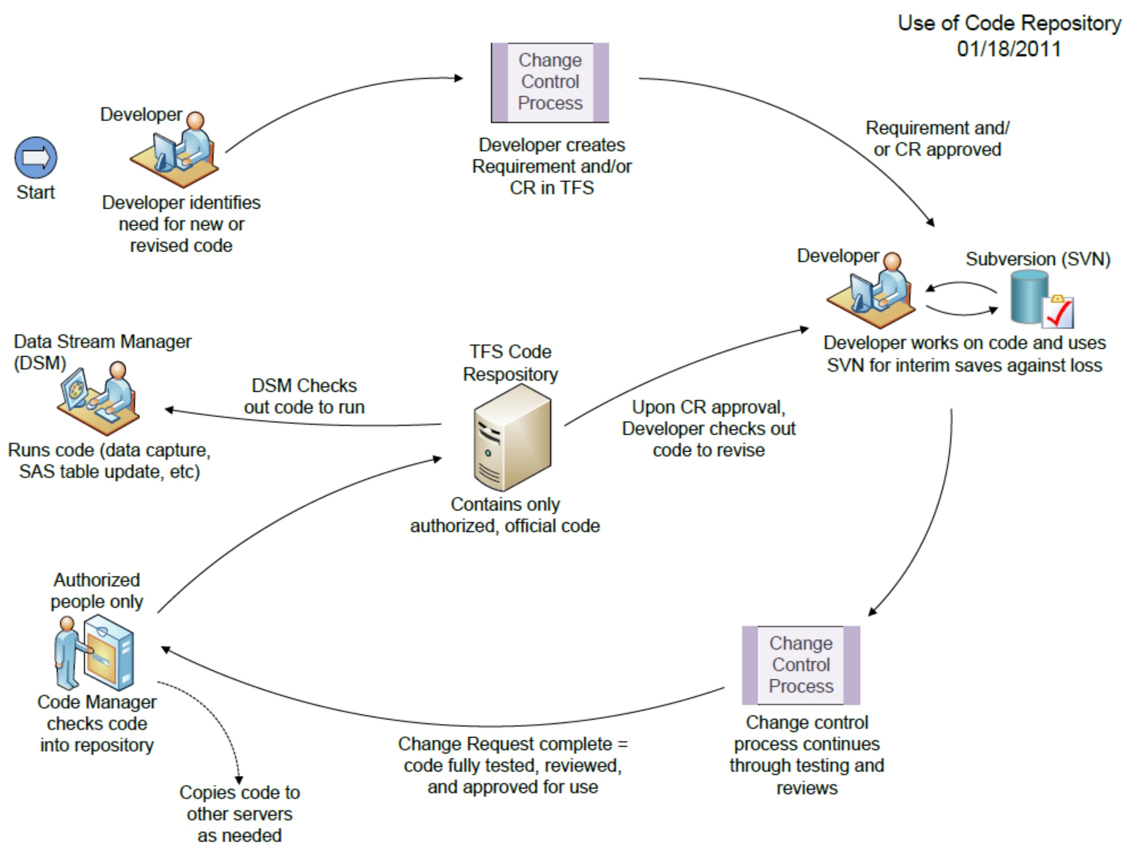


Figure 7. Diagram of the NDMAS change control process used when developing or modifying code or Web pages.

When new or revised code or a Web page is needed, the developer writes the requirements with input from affected parties such as the data stream manager and end user. The requirements are reviewed and approved by an independent data stream manager or developer. For a new code or Web page, the requirements are documented in a TFS requirement work item. For a modification to an existing code or Web page, the requirements for the modification are documented in a change request work item. The developer then creates a change request work item to implement the requirements. The developer designs and creates the new code or Web page. An independent developer then tests the code or Web page to verify that the code performs the functions to meet the requirements. An independent review verifies the testing results and that the change request has been completed correctly. The code or Web page is then passed on to the system administrator to verify that the code complies with NDMAS Standards for Code Development. Once the code or Web page has been verified against standards, the system administrator or software QA checks the code or Web page into the code repository.

Requirements for Web pages are at the Web page level. Requirements for capture, analysis, and display code are for the specific code being developed or modified. The term code refers to a computer program. The term Web page refers to the display accessible over an internet browser. The term software encompasses all artifacts including code, documentation, data, and display.

Step-by-step instructions for completing the change process and Standards for Code Development are on the NDMAS SQA SharePoint page in the SQA Library. Figure 8 shows the SQA SharePoint page, which is reached by selecting the SQA tab from the list of tabs across the top of NDMAS 2.0 SharePoint home page. Instructions for the change request process are contained in the TFS CR Process folder in the SQA library. The TFS CR Process folder is opened by clicking on the 'Name' field. Instructions and diagrams are contained in the folder. Files are opened by clicking on the 'Name' field. Code standards are contained in the folder 'Governing Standards.'

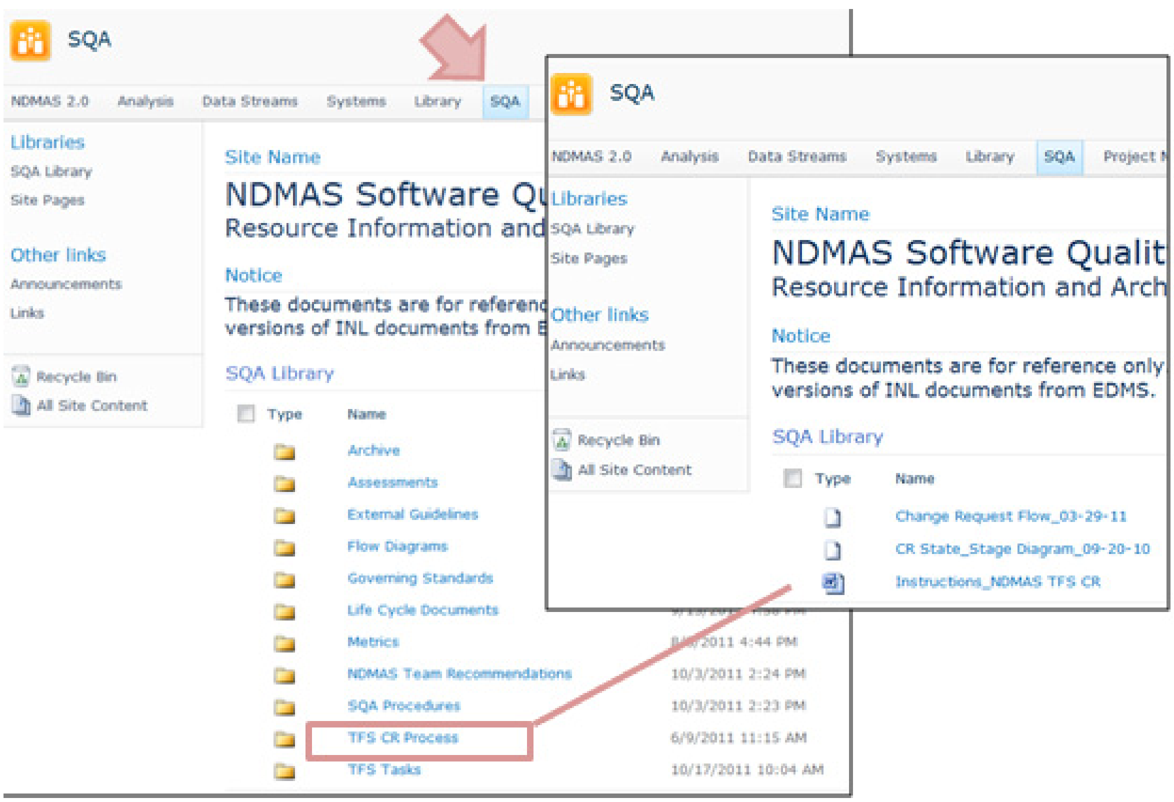


Figure 8. Software QA SharePoint page showing TFS CR Process folder in the SQA library and the instruction files contained in the folder.

5.3 Establish a New Data Stream

Data streams usually have very similar structures for subsequent experiments within a project. The vault has been designed so that data from the same data stream for subsequent experiments from a project go into the same tables in the database. For example, all irradiation monitoring data from the AGR project (experiments AGR-1, AGR-2, AGR-3/4, etc.) go into the same tables. Once a data stream has been established for an experimental sequence, there should be little effort in adding data from the same stream for the next experiment. When a new data stream is first encountered, NDMAS must be configured to assimilate data from that stream. The data stream manager has lead responsibility for establishing a new data stream.

5.3.1 Gather Data for Design

The first step is to gather information from the data generator. Where data are collected following nationally recognized procedures (e.g., ASTM International testing standards), variables, metadata, and reporting requirements may be listed in the procedure. Information needed to establish the data stream is:

- *Documentation*. Get copies of plans, procedures, TFRs, drawings, and other documents that describe the experiment.
- *Variables*. List the parameters that are being measured. How are they being measured? What instruments are being used or what analytical methods? Are there standard procedures that cover the measurements? What are the time spacing, extent, and support?
- *Experiment hardware*. Get drawings of the experiment design. What locations will be needed to describe the experiment? How are the locations related? Is there a hierarchy? What specimens are being measured? What is the specimen numbering scheme? Is there a hierarchy to specimen components?
- If specimens are being irradiated, get details of where the specimens are installed in the reactor core. Determine reference points so that specimens, thermal, and neutronics can be correlated.
- *Units*. List the units the data are in. Are these the units that are desired?
- *Data transfer*. How will data be transferred, at what frequency, what format are the data in?
- *Data type and data category*. As described in Sections 2.1.1 and 2.1.2 respectively.
- *Qualification state*. As described in Section 8.1.2.
- *Verification tests*. Are there any tests that can be applied to the data to evaluate the data such as range tests, correlations, etc? Obtain the values for the tests.
- *Metadata*. List supporting information that needs to be stored with the data. For specimens, what size, shape, material, specimen type, etc? For locations or instruments, instrument properties, location geometry, position, etc.
- *Display*. List the reports, graphs, etc., the user wants to see.
- *Security*. Are there any limitations on who can see the data?

5.3.2 Preliminary Design

Developing the codes and Web pages to process and display a new data stream requires using the change request system to track the review and approval of the codes and Web pages. Use of the change control system is described in Section 5.2. To start development of the new data stream, enter a requirement and associated change request for the following items:

- Capture code
- Database table design
- Extract code, analysis code, cubes, and information maps
- Web pages.

5.3.2.1 Capture Code Design

The design of capture code will depend on the format and timing of the incoming data. There are currently two primary software packages used to capture data into NDMAS: SAS Enterprise Guide (EG) and Microsoft .Net. Microsoft .Net is used for automated data streams from the ATR reactor (RDAS) and reactor experiment monitoring (CDCS). These streams are transferred automatically to a drop box on an INL server. Microsoft .Net monitors the server and executes the capture code when new files are delivered. The RDAS and CDCS files are formatted in nice columns with keywords that allow easy joins to be made to plan tables in the database.

Data streams with poorly formatted incoming data files, or streams with one-time or rare data transfers do not justify developing automated .Net code for capture. SAS EG provides flexible code capabilities and numerous preprogrammed wizards that simplify data capture from a variety of delimited text files and proprietary formats. Numerous examples of capture code can be found in the code repository and development repository to provide guidance on starting a new design. These existing codes contain a wealth of lessons learned from previous data capture activities.

5.3.2.2 Database Design

The tables already exist in the vault for new experiments in existing projects (AGC, AGR). Examine the tables to make sure the existing design will accept the new data stream. If new metadata are being delivered, additional columns may need to be added.

For a new project that closely matches an existing project, use the conceptual designs shown in Figure 5 for time-series oriented data streams or Figure 6 for specimen-oriented data streams as a starting point of the database design. Examine existing tables for guidance on how to design the new tables.

For an entirely new project, start a new design based on the data model concepts presented in Section 3.1. In addition to tables, stored procedures for interacting with the staging and vault databases need to be evaluated and any changes designed to accommodate the new data stream.

5.3.2.3 Deploy and Deliver

The Web page design and the design of the data objects in mart that support the Web page must be closely coordinated. The mart objects must be capable of feeding the desired Web page reports. Therefore, the Web page design should proceed in parallel with the design of the mart objects.

Web Page Design

Based on input from end users, data generators, and review of existing Web pages for similar experiments or projects, design the text boxes, portlets, graphs, and reports to be displayed on the Web page. Determine how to best allow the end user to get detailed information about specific objects. This can be done using OLAP cubes and drill down reports, or having multisection reports and filtering the data that appear in each section, or providing stored processes that return data based on inputs provided by the user. All three methods are currently used.

When available, end user input on how the data will be used is valuable in designing reports. Data can be supplied in tables that are already set up to provide the information needed in the desired format. One use is to have data set up for input to computer models.

Deploy, Cube, and Information Map Design

Once the Web reports needed to display the data are established, the data needed to support these reports can be extracted and formatted. Data is extracted from the vault using SAS programs and transformed as needed. The most common transformation is to pivot the data into multiple columns by response variable. Data that is to be summarized for drill-down reports is processed through OLAP Cube Studio to calculate summary statistics. Information maps are constructed for both cubes and data sets using Information Map Studio.

Web reports are constructed from the information maps and stored in the SAS metadata repository in a separate section for reports that is organized following the organizational structure discussed in Section 5.1. Graphs, text descriptions, and images are stored in an hypertext markup language (HTML) directory that is organized following the organizational structure discussed in Section 5.1.

5.3.2.4 Analysis

The design of each analysis will depend on the needs of the end user and the decisions to be supported by the analysis. If the analysis is to be used repeatedly to support decision making, follow the code development process. Results are either delivered to the end user, or incorporated into the Web page design.

5.3.3 Develop

Development and testing of capture code, extraction code, and Web pages takes place inside the INL firewall. The production environment that the end user sees is outside the INL firewall.

A development version of the NDMAS 2.0 database is used for code testing and development. The development database is on SQL server NDMASDEV and is named NDMAS20, with an accompanying staging database NDMAS20_Staging. The production databases have the same names, but are hosted on SQL server SQL08PRD1. The relevant SAS library references are shown in Table 5. Development follows the change control process discussed in Section 5.2.

Table 5. NDMAS SQL database names and SAS library references.

	Development		**Production**	
	Staging	*Vault*	*Staging*	*Vault*
Database SQL Name	NDMAS20_Staging	NDMAS20	NDMAS20_Staging	NDMAS20
SAS Library Name	SQL NDMAS2 DEV STG	SQL NDMAS2 DEV	SQL NDMAS2 STG	SQL NDMAS2
SAS Libref	NDM2DEVS	NDM2DEVV	NDM2PRDS	NDM2PRDV

5.3.4 Test the Design

The code must be tested by an independent person to assure that the requirements for the code design were met. For data capture, the test must include verification that data that end up in the development vault are equivalent to the raw data provided by the data generator. Documentation of this should be attached to the change request. If manual checking is performed, scanned images of the marked up documents signed by the checker should be attached as PDF files.

The database design should be tested to make sure the data are properly stored and can be associated with the necessary metadata and retrieved to SAS tables of the anticipated design. The data warehouse manager should also test any stored procedures modified to perform verification tests on the data. Web pages should be tested to assure that the reports and stored procedures provide the anticipated results.

5.3.5 Implement the design

Once the code, Web page, and database have been tested, there is an Administrative Review step in the change request process to assure that NDMAS standards for design have been met. The Administrator then moves the code to the code repository. The data warehouse manager implements the database design in the production database vault. The Administrator creates new directories as necessary to store cubes, marts, reports, stored processes, and libraries. The new directories are registered with the SAS metadata repository. These are created inside and outside the firewall.

When the design is fully implemented:

- Capture code will be stored in the code repository
- A directory will exist in the archive to hold the incoming data files
- The database tables will have columns and foreign keys in place to hold the data
- The database will have SQL stored procedures to perform verification tests and to move the data from staging to the vault.
- The database will have views or stored procedures that provide standard presentations of the data and associated metadata.
- Reference tables in the vault will be populated with instruments, locations, specimens, methods, and treatments.
- There will be a library to hold working files for the data stream
- There will be a directory in the OLAP cube storage area.
- There will be a mart for the data stream in the SAS metadata repository to hold production data sets and information maps.
- There will be a folder under the production folder of the SAS metadata repository to hold Web page reports.
- There will be a directory in the HTML repository to hold text pages, graphics, and images for Web pages.
- SAS metadata repository objects will exist both inside and outside the INL firewall.
- Any new security groups for people and Web pages will be established.

5.4 Add Data to the Warehouse

The previous section describes how to prepare NDMAS to receive data from a new data stream and is a prerequisite to this section. This section discusses storing the raw files in the archive, and loading the extracted data into the NDMAS database.

5.4.1 Archive

The data generator delivers the data files to a drop box or to the data stream manager. The data stream manager moves the files to the appropriate directory in the archive. If no changes are to be made to the files, the files do not need to be added to the subversion repository. If changes are to be made to the files, or if at any time in the future changes are made to the files, the files are added to the subversion repository. Section 6.2.7 discusses the use of subversion to track changes to files.

5.4.2 Capture

Capture code is used to extract the data from the raw files, perform the necessary transformations, and load the data into the NDMAS database. The capture process is most often run using SAS EG projects and using intermediate SAS datasets to prepare the data for loading into the database. Other methods can be used. Intermediate datasets created during the capture process are not a VHTR Program record, and can be deleted when convenient.

In addition to loading the data, context information must be added to records. This context information includes:

- Plan ID to associate the record with the measurement being made and what/where the measurement was made
- Event ID to provide time-oriented information about the data
- Data state and data qualification flags
- Data package and batch/session ID to track the origin and processing of the data.

The next few sections discuss the process of adding this contextual information.

5.4.2.1 Role of the Staging Database

All code development and testing takes place in the development environment using the development NDMAS database. Once capture codes are approved, they are moved to the production environment and point to the production database. There are times when it may be desirable to have an intermediate step in the loading of data into the vault. NDMAS provides a staging database in both the production and development environment, where data can be placed after capture but before being loaded into the vault as shown in Figure 9. The staging database, as it is primarily being used at this time, provides a place where accuracy tests can be run on incoming data. Anomalies can be resolved in the staging database before data are moved to the vault. There is no requirement to go through the staging database to the vault. Data received from a data generator that requires no accuracy testing can be placed directly into the vault. Certainly, data that has already been qualified by the data generator before transfer to NDMAS can go directly into the vault. The data stream manager decides whether to use the staging database during ETL. Data are moved from staging to the vault by SQL stored procedures.

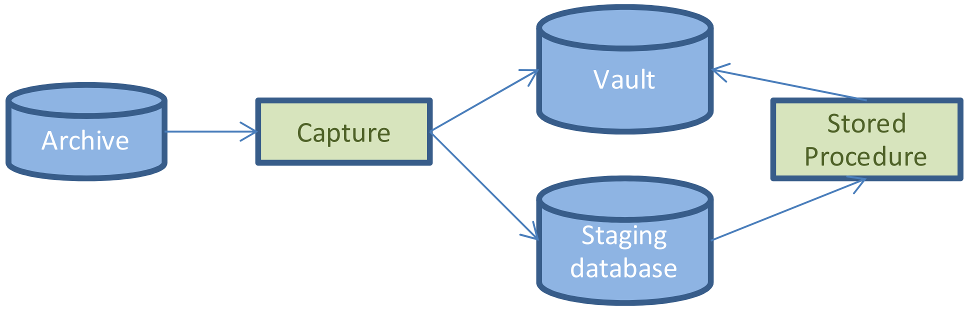


Figure 9. Alternative paths from the archive to the vault are allowed.

5.4.2.2 Assign Plan and Event ID

Information on where the data came from (specimen number, capsule, compact, gas line) is tracked through a foreign key to a plan table. Information on when data were collected is tracked through a foreign key to an event table. All plan and event tables are in the vault database so foreign keys to plan tables and event tables must be obtained from tables in the vault. This applies whether data are being directly loaded into the vault or being loaded into staging. Only data-type tables exist in the staging database. When loading data into staging, capture codes obtain plan and event IDs from the vault, and assign them to the data in staging. Staging does not have foreign key referential integrity for plan or event IDs. This will be enforced when the records are moved from staging to the vault. As part of setting up a new data stream, the reference tables that support the plan tables in the vault must be populated.

Plan ID is assigned by matching the incoming data records to the location, parameter, treatment, and/or specimen the data represent. Events can already exist in the vault or be new events for the incoming data. Code that loads data into staging or the vault must determine if an event record already exists before adding new records to vault event tables. A unique composite index on *StdDateTime* and *SourceDateTime* in all event tables prevents duplicates from being added.

5.4.2.3 Data State

Flags for qualification state and data state are used to communicate the status of data records. The qualification flag will depend on the data Type (see Section 2.1.1 for a discussion of data Type). The assignment of qualification state flags is discussed in Section 5.5. For internal NDMAS processing, the status of data is tracked using the `Data_State_EID` flag. The status of data processing is tracked using the `Verify_Status_EID` flag. The `Verify_Status_EID` flag is discussed in the next section.

The results of NDMAS testing during NDMAS processing of data from native data files to data in the vault is tracked with the use of a flag associated with each record. The NDMAS data state flag can have the following values:

- *Raw*. No testing has been performed and capture has not been verified. It is a short-lived state and, in NDMAS Version 2.0, should not be in the vault. If capture code is tested and approved in the development environment prior to migration to the production environment, this flag should not be necessary. Any data loaded into the database that is considered *Raw* must be loaded into staging for verification. *Capture Pass* is the minimum level for transfer to the vault.
- *Capture Pass/Fail*. Data capture has been verified by showing that data pushed to the database match the raw data provided by the generator. Data captured using approved code are considered verified (capture passed) data. If data fail capture verification, the capture process is reviewed and modified until the captured data are correct. This should usually be done in the development environment, not the production environment. *Capture Fail* should be a fairly short-lived data state and occur only in

staging. For some data streams, capture verification is all that is performed, and data from those streams retain the *Capture Pass* designation in the vault.

- *Accuracy Pass*/*Fail*. For data streams where NDMAS testing and modeling form an integral part of the data qualification process, additional tests are performed on the data to identify anomalies or errors in the data. Data that pass accuracy tests are not flagged; they retain the *Capture Pass* flag value. Data that fail accuracy tests are flagged *Accuracy Fail*. If review determines that an anomaly falls within the expected response of the system and the data are usable, the flag will be changed from *Accuracy Fail* to *Accuracy Pass*. Therefore, data that pass accuracy tests the first time will retain the *Capture Pass* flag. Only data that failed accuracy testing and were later passed will have an *Accuracy Pass* flag. *Accuracy Fail* and *Accuracy Pass* data are pushed to the vault.
- *Obsolete*. Records that have been superseded by newer data revisions but need to remain in the data table are flagged *Obsolete*. Superseded data will be replaced in data tables unless the end user specifically requests both versions be retained. The *Obsolete* flag would be used to identify a data set where, say, an error was made in the calculations and was corrected in the revised data. In this case, however, the data would ideally be deleted. In some cases, new plan entries are created for the new data, and both versions remain active. An example might be two similar computer simulations with different input parameters where both output sets might be considered useful to retain.

Data loaded into the staging database using code that has been developed, tested, and approved using the change request process, and is stored in, and run from, the code repository, are considered verified data. The data state flag is set to *Capture Passed* for data loaded into staging in this way. Data loaded otherwise, have a data state flag set to *Raw*. Other arrangements must be made to verify capture for data not loaded into the database using approved and controlled code. This usually involves manual inspection. Once the capture of *Raw* data has been verified, the data state flag is set to *Capture Pass*.

Manual inspection involves examining referential integrity and manually inspecting a stratified random sample of data. Referential integrity involves verifying all data have been read from the raw data files and that all values can be joined to context information about the data. For example, if a specimen ID is mistyped in one of several spreadsheets transmitting data about a specimen, then a join between data from the different sheets will return a different number of results. The total number of records that is expected should be determined, and the number of records pushed to the database compared to the expected values.

In setting up the strata for the sample, the strata should reflect the structure of the raw data files. All files, all columns in files, and all sheets in workbooks must be included in a stratum. This ensures that if any column, sheet, or file is different from the others, it will be included in the sample. The sampled data values are then manually compared to the raw data files to assure the data in staging are equivalent to the raw data. The printouts are signed by the inspector, scanned to PDF format, and saved in the data stream manager's files. If there is an appropriate change request work item, attach the PDF files to the change request.

5.4.2.4 Verify Status

The status verification flag tracks the status of record processing and communicates to the stored procedures that move data from staging to vault whether the record is ready to be moved. It is only relevant for data placed into the staging database. The verify status for a record must be Success, Pass, or Override to be moved to the vault. Data that are *Capture Pass* or *Accuracy Pass* have `Verify_Status_EID` set to *Pass* or *Success*. Data that fail accuracy tests, but are acknowledged failed data, are moved to the vault. `Verify_Status_EID` is set to *Override* to communicate to the stored procedure that moves data to the vault that the record should be moved even though it is failed.

Two other values for `Verify_Status_EID` indicate that data are not ready to be moved. These values are *Initial* and *Standby*. *Initial* is used when data are loaded into staging that will be tested using stored procedures for accuracy. *Standby* is a general state that indicates data are not ready to be moved.

5.4.2.5 Data Package and Session

Two tracking ID's are assigned to each record in the incoming data and migrated with the data through the staging database to the vault. These are the data package ID and the session (in some places referred to as the batch) ID. As discussed in Section 3.1.2, the data package tracks each batch of data captured to NDMAS back to the data stream and experiment from which it is derived. The session ID tracks the processing of the data through staging and moving to the vault [Note the term `Session_ID` is only used in the staging database data file, all other occurrences of this value are called `Batch_ID`]. A new record must first be created in the vault `rDataPackage` and staging `dCapture_Batch` tables for the incoming data. A data package ID is created with the following information:

- Data package name – derived from the file name(s) or folder name(s) from which the data are read (usually including a date range)
- Date/time the package was captured
- `Person_ID` of the NDMAS operator running the capture code
- `Data_Stream_ID` – ID of the data stream from which the data were generated

A session/batch ID is created with the following information:

- Date/time the batch was captured
- Name of the file(s) the data are read from with full path
- Enum ID of the file type
- Name of the application used to capture the data
- Enum ID for status of data processing, set to same value as `Data_State_EID` (usually *Capture Pass* but could also be *Raw*)
- Enum ID for status of vault processing, set to *Initial*.

Once the ID values are created in `ndmas20.dbo.rData_Package` and `ndmas20_staging.dbo.dCapture_Batch`, the ID values are retrieved and assigned to the incoming data records. Any code that creates data packages and session IDs should first test to see if the package or session already exists. If so, it should retrieve the existing ID and not create a duplicate.

5.4.2.6 Push to Staging

Once all the necessary contextual information and foreign keys are assigned to the incoming data, the incoming data is split so that it lines up with the fields in the data tables in the database. Each table is then filled by inserting the new records into the table. Codes that push data into either staging or the vault must first determine if any of the incoming records already exist in the database, and remove those records from the incoming tables.

Depending on the way data are captured and the need to perform accuracy testing on the data, the suite of flags set by the capture code before pushing the data to the database will be set to different values. Table 6 presents guidelines for setting `Data_State_EID` and `Verify_Status_EID` for data when capturing, testing, and processing data into NDMAS.

Table 6. Guidelines for setting data state EID and verify status flags when uploading data to NDMAS.

Step in Process	Condition	Data_State_EID	Verify_Status_EID
Initial settings when pushing data into staging	No verification	*Raw*	*Standby*
	Captured with code from code repository, no accuracy testing	*Capture Pass*	*Pass* or *Success*
	Captured with code from code repository, accuracy testing	*Capture Pass*	*Initial*
Recording results of accuracy testing	Data pass accuracy test	*Capture Pass*	*Pass* or *Success*
	Data fail accuracy test because StdDateTime > FailDateTime	*Accuracy Fail*	*Override*
	Data fail range test, incoming value missing, or other type of test	*Accuracy Fail*	*Failed*
Recording results of data stream manager review of failed data	Data stream manager acknowledges failed data	*Accuracy Fail*	*Override*
	Data qualification review accepts data as accurate and reverses results of test	*Accuracy Pass*	*Pass* or *Success*
Initial settings when pushing data into vault	Captured with code from code repository, no accuracy testing	*Capture Pass*	Not Applicable

5.4.3 Accuracy Testing

Activities in the staging database include:

- *Verify capture*. Captured data must be equivalent to the original raw data. If data capture is not performed using code that has been tested and reviewed and run from the code repository, then another means of verification is necessary. Data loaded into staging without using an approved code must be given a data state of *Raw* and a verify status of *Standby* so that it is not processed to the vault. Verification would then usually be performed by manual comparison of captured data to the original raw files provided by the data generator. The better way to capture data is to develop code to perform the import, transformation, and storage using the development database. The code is tested and reviewed to verify that code performs correctly. The code is then stored in the code repository under configuration control. Use of tested and controlled code provides verification of data capture. Data captured and loaded to staging using approved code is assigned a data state of *Capture Passed* and a verify status of *Initial*. This tells the stored processes that the data are ready for testing. Data captured using an approved code can also be loaded directly into the vault if no testing is required.
- *Perform data range checks*. Testing for data out of range screens for values that are beyond the limits expected for the data acquisition system. Some of these are due to electronic gremlins in the system. Other times, the data out of range indicates an upset in the process being monitored or a failure of the instrument collecting the data. Data out of range is reviewed with the data generator to determine how to disposition the anomalous data.
- *Perform failure/trend tests*. Instruments in the harsh environment of the core of a nuclear reactor fail. Quite a few thermocouples have failed in the AGR experiments. When an instrument has failed, the date of the failure is stored in the database plan table for that instrument. This test compares the data collection date to the date the instrument is known to have failed. Any data collected after that date are not considered to represent conditions in the reactor.
- *Referential integrity checks*. All of the incoming data should align with entries in the plan tables in the vault. Data should be coming from known locations or known specimens. Any failure to have a

complete match between incoming data and entries in the plan tables indicates an error in referential integrity. This may be because of an error in entering a specimen ID. Referential integrity issues must be resolved before data can be pushed to the vault.

Stored procedures are used to perform any accuracy tests to be run on the data in staging and to assign data state flags to the data based on the results of the tests. Not all data are subjected to accuracy testing. Accuracy testing is mainly applied to RDAS and CDCS feeds from irradiation experiments. Plan tables in the vault contain four fields to support accuracy testing. These represent the minimum and maximum expected values for an instrument or parameter, the date/time that an instrument failed, and the date/time that an instrument started giving unqualified data that are still considered useful (*Trend* data). Stored procedures are run that examine these fields in the vault, and if values exist in these fields, compare the incoming data to those values. Data that pass the accuracy tests retain the `Data_State_EID` of *Capture Pass* and are assigned a `Verify_Status_EID` of *Pass* or *Success*. Data that fail these accuracy tests are assigned a data state of *Accuracy Failed* (Table 5). Failed data are data that fall outside the expected range of values or were collected after an instrument failed.

Data collected from instruments that have failed are considered to be acknowledged failed data. The verify status is automatically set to *Override*, and the data are moved to the vault. For data that fail by range test, or because the incoming value is missing, or other tests, are considered to be unexpected failures, and are not automatically moved to the vault. The data stream manager must review the failed records to determine whether any corrective action needs to be taken.

5.4.4 Managing Accuracy Failed Data

Data that fail range tests, or because the incoming value is missing, or other tests, are considered to be unexpected failures. If the data stream manager finds that no corrective action is needed, the verify status flag is changed to *Override*, leaving the `Data_State_EID` flag at *Accuracy Failed*, and the data moved to the vault. An example of when this might occur is when a single temperature value is recorded well outside the range of plausible temperatures (like -2,000,000°C). This occasionally occurs and does not require corrective action, just acknowledgement that a value out of range was recorded. Another example, the N16D flag was not reported along with ATR power data for the first several cycles of AGR-1. A missing N16D flag for these data is an expected occurrence, and the data stream manager would override the failure of ATR power data. However, new data that arrives with a missing N16D flag signals the data stream manager to pursue the missing flag to determine why the incoming data stream was changed.

Data that failed range tests during data qualification may be deemed to be an accurate representation of conditions in the experiment capsule. Moisture in the gas flow exiting the AGC capsules is an example of this type of situation. The moisture specification for AGC irradiation experiments is <10 ppmv. Moisture at times exceeds 10 ppmv. This may occur because gas flow to the capsule was shut off for a period, or early in an experiment when the capsule was drying out. In the former case, the moisture measurements do not reflect conditions in the capsule because there is no sweep gas flowing to bring moisture from the capsule to the sensor in the outflow gas line. These data would be considered failed data. In the second case, the high moisture values do represent moisture being flushed from the capsule, and these data would be passed. When previously failed data are reclassified as passed, the `Data_State_EID` is changed from *Accuracy Fail* to *Accuracy Pass*. The `Verify_Status_EID` is changed from *Fail* to *Pass*. The data are then moved to the vault. The data may already be in the vault when the status is changed. In that case, the `Data_State_EID` flag in the vault is changed.

5.5 Moving Data from Staging to the Vault

SQL stored processes created and managed by the date warehouse manager are used to move data from staging to vault. These processes run on a schedule, and respond to the value of

Verify_Status_EID. All data with a *Pass* or *Override* flag will automatically be moved from staging to vault (copied from staging to vault and deleted from staging). Data with *Failed* Verify_Status_EID will be left in staging. The data stream manager of a data stream that uses staging is responsible for monitoring staging for failed data, and taking action on any data left in staging. Pending future funding, tools will be created to assist data stream managers with reviewing and managing data in staging. For the time being, this must be done manually using SQL Server Management Studio or SAS Enterprise Guide. The date warehouse manager is responsible to move data from staging to vault as covered in SDD-228, "NDMAS2.0 Database" design document.

5.6 Manage Qualification State Flags

NDMAS assigns Qual_State_EID to indicate the qualification status as defined by MCP-2691 and listed in Table 7. *Type B* and *Type D* data are not qualified, so this flag is set to *Not Examined* for those data types. For time critical *Type A* and *Type C* data that are displayed on the Web before final qualification states are determined, the flag is set to *In Process* so that data users understand that the data have not been through qualification. The qualification status for *Type A* or *Type C* data that have been qualified prior to submittal is set based on the outcome of the existent review. Once verification against requirements is complete, qualification states are assigned per MCP-2691 as *Qualified*, *Trend*, and *Failed*. *Failed* records are not displayed to users. Records in the *Trend* and *Qualified* states are displayed to users with the flags associated with the data to communicate the state of the data to the end user. Definitions follow MCP-2691 and are discussed in Section 8.1.2. NDMAS retains all data no matter what the qualification state flag. *Failed* data, particularly from irradiation monitoring experiments, can be useful for analysis purposes.

Table 7. Assignment of qualification state flag based on data type and data category.

Step in Process	Condition	Qual_State_EID
Initial settings when pushing data into staging	*Type B* and *Type D* data	*Not Examined*
	Type A and *Type C* data prequalified by the data generator	*Qualified, Trend, Failed* as assigned by the generator
	Type A and *Type C* data, time critical, but not qualified by the data generator	*In Process*
Moving data from staging to the vault	All data	Retain the flag value set in staging.
Recording results of data qualification reviews in the vault	Data meet requirements for use	*Qualified*
	Data have some problems, but are still useful	*Trend*
	Data do not meet requirements for use	*Failed*

5.7 Revise Data in a Stream

Data revisions are incorporated in the archive and the vault. Archive changes are documented using Tortoise SVN software; vault corrections are documented using a change control procedure/system. Wholesale replacements are not tracked in the vault. Changes are not tracked in mart or on Web pages.

A replacement occurs when calculations are redone with revised parameters and the revised dataset is sent by a data generator. This has occurred several times in the past. When these new files are received, all the data must be reentered into the database. Corrections are for subsets of the data, either a few numbers, or one column, or a few time periods of data.

Because date/time is held in a table separate from the data, the data stream manager must evaluate the impacts of the revisions or changes on the event table and the foreign key links from the event table to other data tables. If the new data are all at the same date/time values as the previous data set, the data values can be replaced or updated using an existing Event ID.

5.7.1 Corrections

If the data generator corrects the raw data, the changes are documented in the archive. The corrections do not have to be made to the raw data files, but a copy of the document transmitting the corrections is stored in the archive where the original files are stored. It is preferable to get a corrected file so that the archive data are correct. Since this is a change to a VHTR Program data record, Form 435.78 should accompany the corrections.

A stored procedure or SAS code is written to copy the affected records from the production vault to the development vault database. Using the change request process, the developer develops the code to make the necessary corrections to the data. The data in development vault are compared to the corrections to test the update code. If the update is correct, the update code is used to update the incorrect data in the production vault using the tested and approved code.

5.7.2 Replacements

If new data files are transmitted to replace existing data files, a new Form 435.78 indicating a revision number must accompany the new data files. If the files are meant to replace existing files, the old files in the archive should be replaced by the new files. Since deleting a file in SVN removes the file from all future versions in the repository, it will not show up in future updates. However, the file will continue to exist in older versions of the repository when the file did exist. It can be found by searching older revisions. To replace data already in the archive:

- If the new files have the same name as the old files, add the old file to the repository, if not already added, and commit; write the new files over the old files and commit. This will track changes in the files between versions.
- If the new files have different names, but are really the same (like have a “rev1” included in the name) rename the files so they match the old files, and overwrite the existing files.
- If the files are not one-to-one replacements, delete the existing files (after making sure they have been added and committed to the subversion repository) making sure a good description is included in the notes section of the SVN dialog box to make these files easy to find. Move the new files to the archive, add them to the repository, and commit.

Capture codes and the vault are configured to prevent duplicate records from being entered into the vault. Therefore, if new records are to replace existing records, two options are available. The new records can be used to update existing records in the database with new values, or the old records can be deleted and new records added. Very little updating has been used in the past, but should be used when appropriate rather than deleting and replacing records. Because event tables are shared by multiple data tables, deleting event table records can usually only be done if every variable is being replaced. If the decision is made to delete rather than update, it is probably better to delete existing data from the data tables, leaving the plan and event tables intact. The capture code will join the incoming locations and events to existing records in the vault, and assign the appropriate plan ID and event ID to the incoming data.

If the end user wants to keep the older version of the data, the new data will be inserted into the database. The new data records will be made unique from the existing data records, or the new data will not be added to the vault. Therefore, to keep two versions of the same data in the vault, something like the Plan table or the Treatment table will have to be modified to make the new records unique.

5.8 Collaboration

To facilitate collaboration among NDMAS team members, NDMAS maintains a collaboration site run under Microsoft SharePoint. As currently configured, the SharePoint site contains pages for Analysis, Data Streams, Systems, Library, Software Quality Assurance (SQA), and Projects. A screen capture from the Data Streams page is shown in Figure 10. SharePoint offers a wide range of tools that support collaboration. The NDMAS team uses these tools most often. These tools are described as follows:

- *Libraries*. Collections of reference documents, articles downloaded from the Web, and documents written by the NDMAS team are stored in on-line libraries. For INL controlled or record documents, the official versions of the documents are in EDMS. The SharePoint libraries contain a link to the record copy in EDMS where that is available. Libraries provide a one-stop shop for finding references.
- *Lists*. Lists of links to tips and how-to's that people find on the Web. This allows sharing knowledge among the team.
- *Document preparation and review*. SharePoint has tools for group preparation and review of documents. When a new document is being prepared, a document workspace is created for the duration of that effort. The completed document is then stored in a library on one of the Web pages.
- *Discussion groups*. Forums are used to hold discussions among team members to reach consensus on how to do business. The forum then holds a history of the discussion and conclusions drawn. This is much more traceable than a flurry of emails.
- *Training*. Libraries provide a source for new team members to go to find information on the NDMAS system.

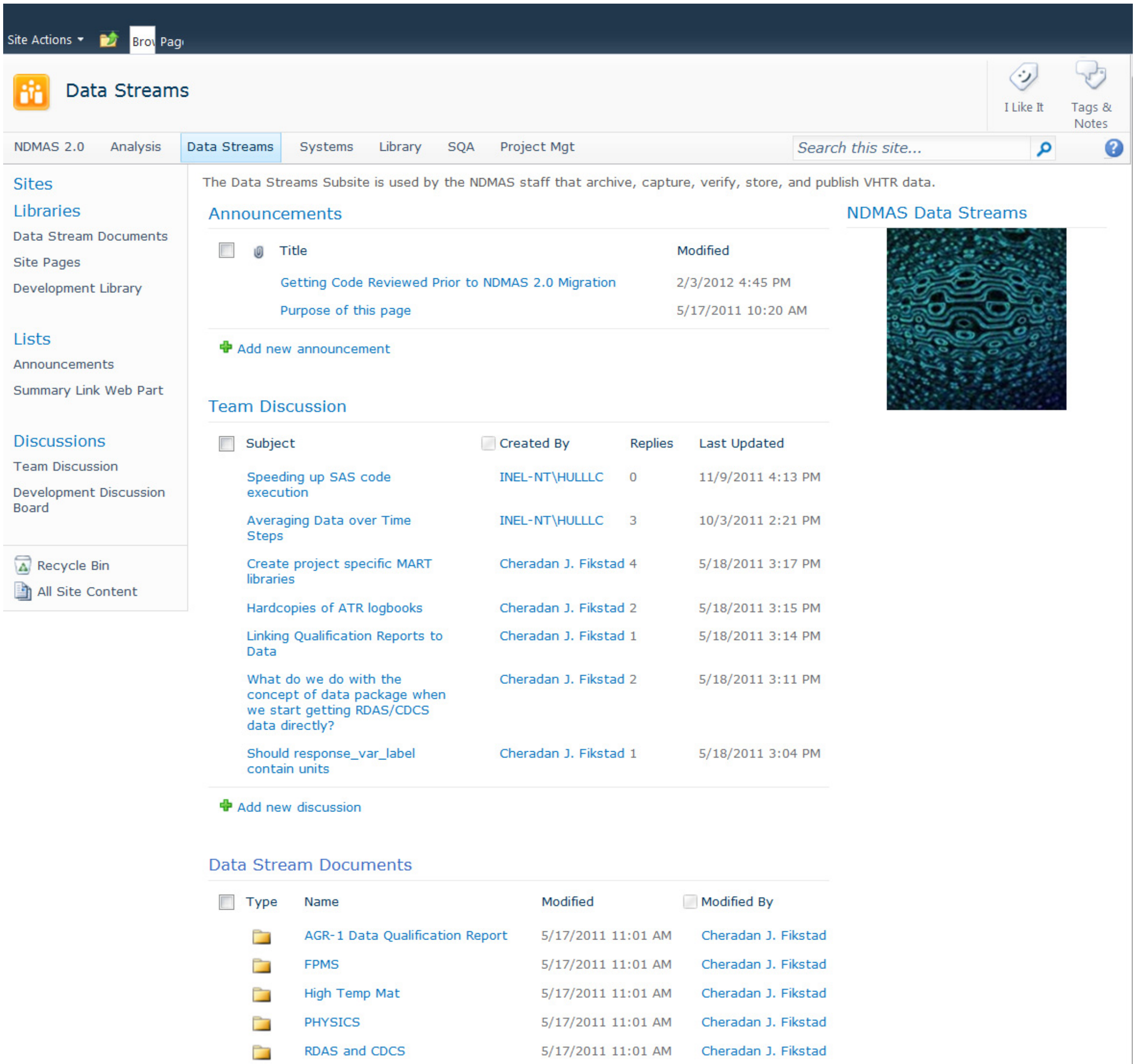


Figure 10. Screen shot of the Data Streams SharePoint page showing announcements, discussion forum, and document library Web parts.

6. SYSTEM MODEL

The NDMAS team of developers works inside the INL firewall to perform ETL, code development, and Web page design. End users of the NDMAS system are from the INL and other institutions, including foreign countries. Delivery of data occurs outside the INL firewall to allow data access for all users, including those from outside of INL. This section describes the hardware and software that comprise the NDMAS system.

6.1 Hardware

All of the servers NDMAS uses are INL network servers. Two of these have special significance, the NDMAS and SQL servers shown in Figure 11. INL network servers provide drop boxes for transmitting data to NDMAS along with backup of the NDMAS server and SQL server to provide loss prevention of files. Subversion repositories are also located on INL servers. The NDMAS server contains the archive and SAS metadata repository for NDMAS. It is also the location of the code repository. The SQL server houses the NDMAS vault and staging databases. NDMAS maintains a server outside the INL firewall to host the Web pages along with a duplicate SAS metadata repository.

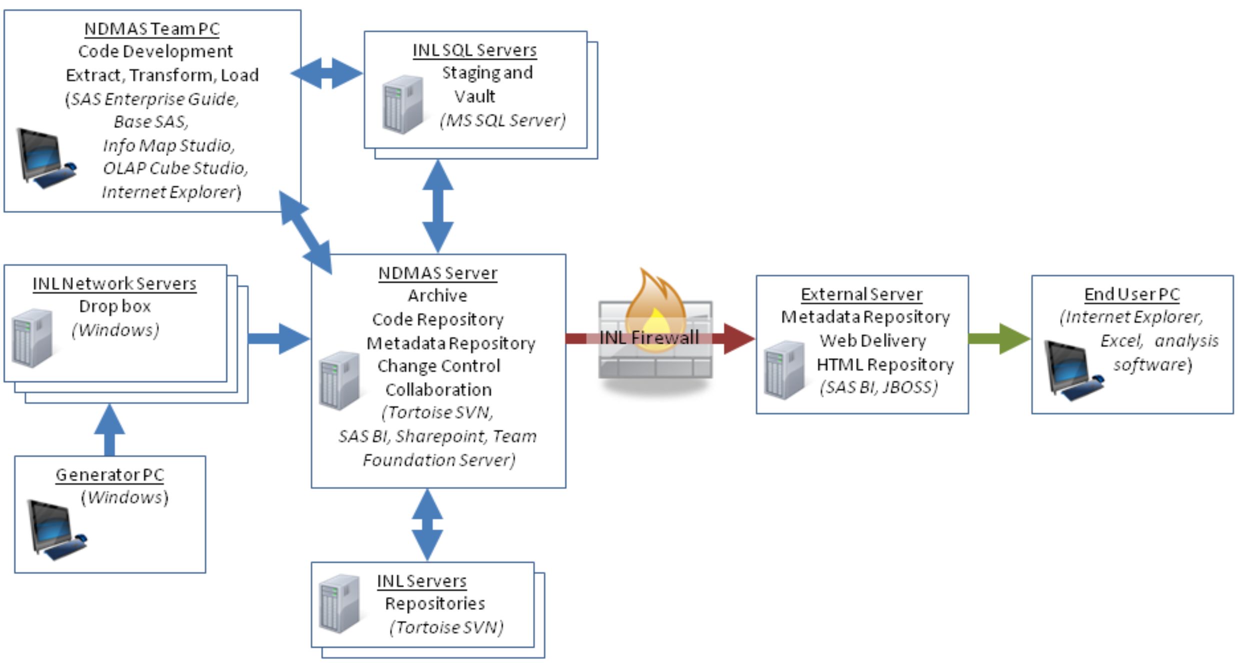


Figure 11. System model of the NDMAS system showing the hardware, NDMAS components associated with the hardware, and software associated with the hardware and components.

6.2 Software

The NDMAS system relies heavily on SAS Business Intelligence software. The mart and Web delivery system are all powered by SAS software. The bulk of the ETL is performed using SAS Enterprise Guide.

6.2.1 SAS Metadata Server

SAS Metadata Server is multiuser software that administers repositories of data and data-related objects. The mart and Web delivery systems of NDMAS are based on the SAS Metadata Server. There are two realms for the mart: the data-object realm and the data-related-object realm. The data-object realm, which is based on the physical file system on the server, contains SAS datasets, Web report XML files, SAS stored process codes, cube definitions, and other physical files located in a physical file directory tree. The data-related-object realm stores information about the physical objects. This would include information on column names, data types, length of character variables, formats, etc. This information about the objects is referred to as metadata. The metadata is stored in a folder system that echoes the physical directory system, but can only be accessed through SAS software. The physical file contents and the metadata about the file contents must be kept in sync so that SAS can access the contents of the physical files. Proper organization, use, and maintenance of the SAS metadata repository are essential to maintaining a functional SAS EBI Platform operation.

The common SAS-related objects used by NDMAS are listed in Table 8 along with descriptions, examples, storage locations, and path to the SAS metadata about the object. Generally, data stream managers and analysts do not worry about physical file location. Objects created through SAS, such as a SAS dataset created by an EG Project, are saved by using the SAS library name. SAS places the file in the appropriate physical file location based on the library reference. Access to that file in the future is accomplished through the library reference, and the physical location of the data set is not relevant to the person using the data. To find the data, the analyst accesses the data through the SAS library structure or through the SAS metadata folder structure.

6.2.2 Management Console

SAS Management Console provides a single point of control for managing the NDMAS intelligence creation process, including data integration, storage, reporting and analytics. By leveraging one administrative interface, the console requires less training time, reduces the number of steps involved in administration and provides an opportunity to uphold standard operating procedures and minimize manual work by enabling repeatable processes.

Management Console can be run from an administrator's desktop system, yet it can manage the resources on all platforms supported by SAS. It provides centralized management of:

- *Resource access control policies*. Provide the ability to grant access to resources by users or groups and assign attributes to protect those resources.
- *Library and server definitions*. Logical names for libraries and servers can be set up by the administrator.
- *Metadata repositories*. An administrator can manage the creation, replication, and promotion of repositories from his or her desktop, eliminating the need for the administrator to be located near the servers.
- *Server administration*. An administrator can monitor and administer all servers and sessions on those servers within the SAS environment.
- *Technical and business metadata*. Provides centralized organization and management of all significant business and technical metadata. The console also works with XML-generated metadata, predefined processes and tasks, and business metadata.

Data stream managers, developers, and analysts can access a subset of Management Console functions from their desktops to perform maintenance task on data, such as synchronizing metadata definitions and adding descriptive comments to SAS objects.

Table 8. Descriptions and storage information for SAS objects and metadata about the SAS objects.

Object	Contains	Use	Example	Physical File Location	Metadata Folder
Library	SAS dataset	Working files with retention > current session (otherwise use work library)	AGC1_IRR AGR2_PIE	D:\SAS_Projects\NGNP\Library\ [Experiment]\[Stream]	/Shared Data/NGNP/Library/[Experiment]/[Stream]
Library (Mart)	SAS dataset	Holds files that feed Web reports, cubes, information maps, graphs etc for display to end users; also reference data sets used by multiple analysts	NDMAS AGC1 Mart NDMAS HTM Mart	D:\SAS_Projects\NGNP\Library\[MARTS]\[Experiment]\[Stream]	/Shared Data/NGNP/Library/Marts/[Experiment]_MRT
Library (SQL)	Open database connectivity connection to SQL database	Secure storage of VHTR data in a relational database	SQL NDMAS2 SQL NDMAS2 STG	INL server SQL08PRD1	/Shared Data/NGNP/Library/SQL NDMAS2
Cube	Summarized multidimensional, hierarchical data	Rapid analytical processing of large multidimensional data sets with ability to drill down for more details		D:\SAS_Projects\NGNP\Cubes \[Experiment]	/Shared Data/SASApp-OLAP schema
Information Map	Descriptive information about data and data relationships	Provides labels, descriptions, filters, joins, and security for data in SAS datasets or OLAP cubes.	Name should include name of cube or table	Managed by SAS	/Shared Data/NGNP/Library/Marts/[Experiment]_MRT (stored in same Mart as data)
Web Report	HTML code to display data, graphs, images or text on Web pages	Deliver VHTR data and analyses to end users		Managed by SAS	/Shared Data/NGNP/Production/[Experiment]
Stored Process	SAS code	SAS program along with metadata and security authorizations; can be accessed from a variety of clients.		D:\SAS_Projects\NGNP\STP\[Experiment] (source code repository, restricted write access)	/Shared Data/NGNP/Production/[Experiment]

6.2.3 SAS OLAP Cube Studio

SAS Online Analytical Processing (OLAP) Server is a standards-compliant OLAP data source that uses multidimensional expressions (MDX) to query and navigate through multidimensional information. MDX is a structured query language similar to SQL. The central component of SAS OLAP Server is a multithreaded query engine that ensures optimal use of hardware resources. Any query sent to the server is handled by an individual query thread, enabling the server to handle large user communities accessing the server in parallel. OLAP data sources can be spread across multiple file systems, enabling the multithreaded query engine to execute multiple queries in parallel by reading data from multiple locations at once. The MDX query engine accesses data stored in external relational tables (ROLAP), SAS' own multidimensional format (MOLAP) or a combination of data stores (HOLAP). OLAP source designers are free to choose the format that best supports their business requirements.

Routines are included that enable SAS procedures to query the SAS OLAP Server using SQL Pass-Through with reach-through from multidimensional data to underlying detailed data. Individual summaries inside OLAP data sources can be stored in a compressed format if they are not going to be frequently accessed. Data compression significantly reduces the space required to store information. Bundled with SAS OLAP Server, SAS OLAP Cube Studio provides cube designers with an easy-to-use graphical user interface for creating OLAP data sources. Data designers can also define OLAP data sources with a programmatic approach using a SAS procedure. A Java-based wizard facilitates designing, creating, updating and tuning cubes. The wizard is available through SAS OLAP Cube Studio.

All definitions of OLAP data sources are stored in a centralized repository on the SAS Metadata Server. The SAS Metadata Server can significantly reduce the costs of running a SAS environment because it provides a single point where administrators can maintain the entire platform and a central point where applications can search for information about available data sources. The SAS Metadata Server also maintains central security information required to prevent unauthorized access to information. Access rights can be set for entire OLAP data sources or for individual dimensions using user- or group-specific conditions.

The SAS OLAP Server owns a cache that stores frequently accessed cubes in memory. Once a query accesses a cube, the cube's regions affected by the query remain in memory, enabling subsequent queries to access these cached regions quickly in memory without having to access the slower disk-based storage. Dimensions can be shared among cubes for consistency and efficient updates. An OLAP administrator can define a set of dimensions once and then reference the set from other cubes. This saves processing time and storage space, as separate dimensions are not needed for each cube. Administrators can also update the shared dimension once and apply it to cubes sharing that dimension. System administrators can use performance reports generated by SAS OLAP Server's Application Response Measurement (ARM) system to fine tune cache sizes and cube structures according to the usage patterns of an OLAP data source. SAS OLAP Server can be monitored through ARM classes. ARM result sets can be transferred to SAS or third-party clients, such as the HP-UX Workload Manager, for monitoring activity and performance.

6.2.4 SAS Information Map Studio

SAS Information Map Studio is a desktop Java application that enables information architects and query designers to build information maps–a business metadata layer that describes the physical data warehouse. This layer distributes warehouse data to business users in terms they understand, providing self-service business intelligence capabilities. These metadata definitions are presented to business users through SAS reporting interfaces, enabling them to query the data without requiring technical knowledge and giving them confidence in the results they retrieve.

Through SAS Information Map Studio, you can create data items that map physical data references to business-context terms. Creating a business metadata layer translates complex data structures into terms that business users can understand, enabling consistency and self-sufficiency. Information maps can contain information about data sources, data relationships, and data items and their usage. The data items can be easily organized into folders and subfolders to help users quickly find the information they need.

Information maps can surface standard definitions for business metrics so all users can access the same definitions. These business rules ensure that key business metrics are defined and used consistently across the enterprise. Information maps also capture proper usage information, such as allowable aggregations. This usage information sets boundaries for business users to ensure that data is used only as intended and that business users have the appropriate information on how to use a data item.

SAS Information Maps are stored in the central SAS metadata repository so they are available consistently across the suite of SAS reporting interfaces and to SAS solutions built on the SAS Enterprise Intelligence Platform. Information maps are integrated across the entire SAS Intelligence Value Chain so metadata maintenance is no longer an issue.

Information maps can be organized into a folder tree in the SAS Metadata Server, and the SAS Information Map Studio can be used to create, move, rename or delete the information maps. A query testing facility enables users to view and validate the query code (either SQL or MDX), based on selections from the information map.

SAS Information Map Studio allows the applications of row-level permissions to an information map based on relational data. As a result, data is filtered for specific user groups so each group sees only its authorized subset of the data. Individual information maps, as well as folders containing multiple information maps, can be secured from within SAS Information Map Studio or by using SAS Management Console.

6.2.5 Web and Desktop Reporting

SAS Enterprise EBI Server enables users to quickly create basic queries and reports. Users can view reports in a self-service manner while NDMAS maintains control of the underlying data and security. Data is presented in terms users understand so nontechnical users can search and choose the information they need. As needs evolve, more sophisticated layout and query capabilities are available. Features include:

- A Web-based, interactive reporting interface for information consumers
- Gallery of common, predefined layouts and custom templates
- Build, load, organize, view and save reports based on OLAP cubes and/or relational data from one or more data sources
- Provide multidimensional data exploration: drill, rotate, filter, reorganize, sort, toggle totals, export to various target formats and more
- Provide relational data exploration: rotate, filter, reorganize, sort, toggle totals, export to various target formats and more
- Print reports to PDF or export formatted tables and graphs or data to Excel
- Deliver a Web-based, interactive report-building interface for report authors
- Create comprehensive reports with powerful layout capabilities for experienced report authors and specialized reports
- Add hypertext links on text, images, tables, graphs and group breaks, reports or Web pages

- Out-of-the-box capabilities such as report linking within report tables and the ability to incorporate custom images within report tables
- Support for embedded HTML in data items, such as images and hyperlinks, in tabular output
- Link to a specific report section within the same report
- Use conditional highlighting on tables and graphs to define exceptions
- Use a variety of charts: bar/3-D bar with multiple lines, pie/3-D pie, line, scatter and tile charts with the ability to add annotated reference lines to graphs
- Save predefined reports and information with report archiving
- Common prompting framework allows for the creation or reports that allow prefiltering based on the data, eliminating the need for constant recreation of reports and filters as new data items are added
- Search-in filter allows users to quickly filter large amounts of detailed data to see just what they need in an efficient, productive manner
- Multiple repositories and content location independence
- New role framework for customizing existing roles or creating new roles to authorize capabilities.

6.2.6 SAS Enterprise Guide

Most of the data displays and statistical data modeling generated by NDMAS are created using SAS EG. This product provides an environment for accessing data, building queries that provide desired views of the data, including adding and removing fields and filtering the data, and performing a series of statistical analyses. The product also provides tools to graph the data and the model outputs.

Enterprise Guide comes with the SAS Business Intelligence software package. It is installed on the SAS Next Generation Nuclear Plant (NGNP) server. NDMAS team members access EG using a client interface on their PCs. EG also provides access to the underlying SAS Foundation, which includes an open-ended language for manipulating data and invoking SAS statistical procedures. The Base SAS product also includes a macro language that allows repetitive processes to be easily automated.

6.2.7 TortoiseSVN

Subversion is a multiplatform, open source version control system. It consists of a repository database and a graphical user interface. Subversion is primarily a system to manage code development among multiple software developers allowing users to track changes, restore previous working versions, and resolve conflicts among multiple versions of the same code. These same tools allow tracking changes in any file, with the ability to roll-back the file to previous versions. NDMAS uses subversion for code development and tracking changes to raw files in the archive.

TortoiseSVN is open source software and is available from sourceforge.net. NDMAS Team members developing code or working with the archive need to download and install the software on their PCs.

To develop code, the developer creates a directory on their PC to hold the development files. While the name of this directory can be anything, it is less confusing to name it the same as the name used on the repository server. The checkout can occur at any level in the directory tree, so only check out those directories that are needed. Right click on the empty directory to check out a copy of the repository. This creates a working copy of the repository on the local PC as shown in Figure 12. Using the TortoiseSVN GUI interface, the developer keeps the codes on their local PC in sync with copies in the repository.

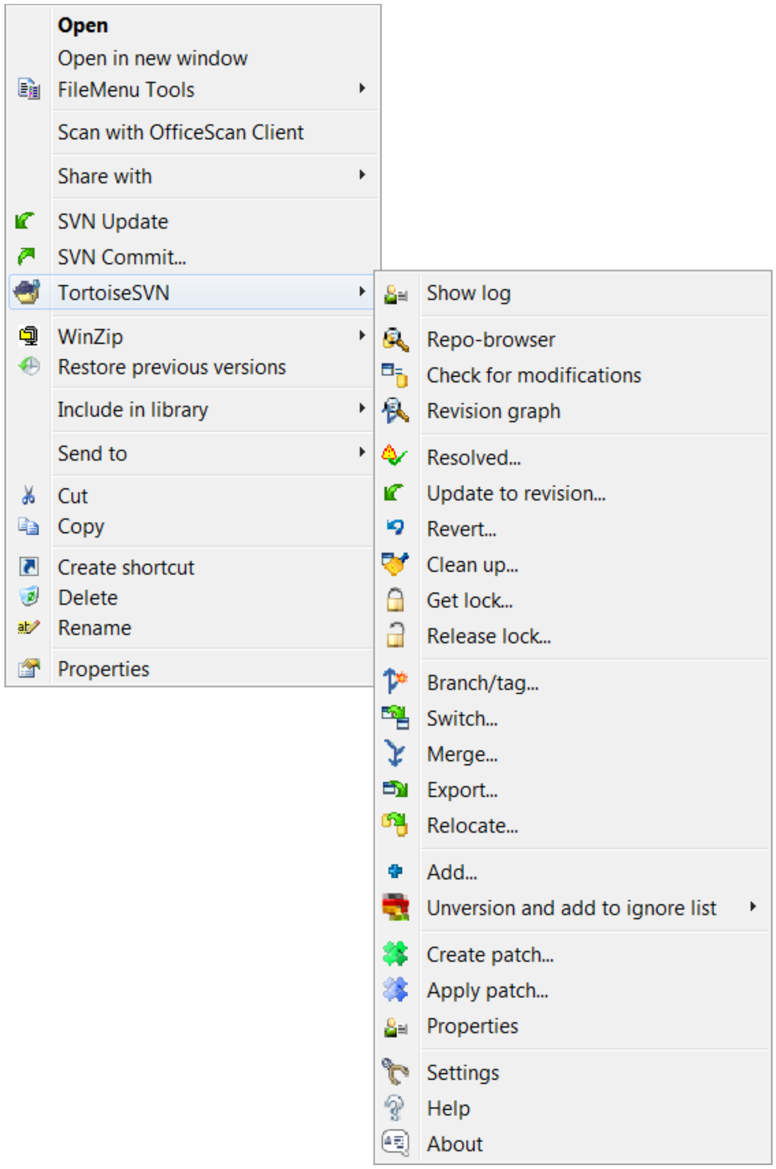


Figure 12. The TortoiseSVN menu is reached by right-clicking on a file in Windows Explorer. SVN options are added to the menu when Tortoise is installed on the local PC.

The URL of the entire repository is https://svnserver.inl.gov:9443/repos/NGNP_SAS/.

The developer needs either or both of the following directories for code development:

- NDMAS_Version_1_1_SAS_9_2 – code designed to work with the NDMAS Version 1 database
- NDMAS Version 2.0 – code designed to work with the NDMAS Version 2 database.

Over time, all active code should migrate to the Version 2.0 directory. The developer develops code on their local computer, periodically committing updates to the repository. For new code, the developer saves the new code file in the local working copy of the subversion directory, uses the TortoiseSVN interface to add the file to the repository, then commits the contents of the file.

The archive has a separate subversion repository. The working copy of the archive repository is on the NDMAS server in the directory \NGNP_DATA. Do not create a copy of this repository on your PC.

For the archive, files that are not changed do not need to be added to subversion. Only files that are modified are added to the subversion repository. Subversion is not a backup of the archive. Backup of the archive is performed separately. Subversion is to create a traceable history of changes to files. NDMAS Team members moving raw data files into the archive and capturing data are responsible for adding incoming files to subversion if changes are to be made to the files.

6.2.8 SharePoint

SharePoint is a Microsoft platform that allows people to build Websites (Sagi 2011). The software contains many components that allow construction of custom pages for specific purposes. The available Web parts are particularly well suited to team collaboration. SharePoint sites have many built in features that make them useful, flexible, and customizable such as: security management, lists, document libraries, discussion forums, calendars, and announcements. The NDMAS team maintains a SharePoint site to store document libraries, lists of links to useful Web sites, discussion forums, and to host document preparation work spaces. The SharePoint site is where NDMAS team members can go to find style guides, step by step guidance to NDMAS processes, and links to program and project QA documents. The home page of the SharePoint site is on the INL intranet at https://nteam/sites/NDMAS/default.aspx.

7. SECURITY AND ACCESS TO THE SYSTEM

NDMAS system information and computer security are based on industry standards and the capabilities of the software solutions used. Security is applied at multiple levels. NDMAS team and system administration personnel have access to the data warehouse and various repositories used by the team as follows:

- *Code repository*. Only administrators and software QA personnel have write access to the code repository. Data stream managers and developers have read access to the code repository.
- *SAS metadata repository*. Only administrators have authority to modify the structure of the repository. Data stream managers and developers have permission to create, read, modify, and write files within the repository structure.
- *Archive*. Data stream managers can modify the structure of the archive as well as create, read, write, and modify files within the archive.
- *Vault*. The data warehouse manager has rights to modify the structure of the vault. Data stream managers can read, write, and modify data within the vault following NDMAS team processes for controlling the quality of data in the vault.
- *Mart*. Administrators have access to modify the structure of the mart, which is part of the SAS metadata repository. Data stream managers and developers have read, write, create, and modify permissions for files within the mart.

7.1 Credentials

User access to data is provided by way of Web pages. Access is managed through a unique set of credentials for each user. There are two separate sets of credentials, depending on whether access takes place inside or outside of the INL firewall. Development and test environments are internal to the INL network. Credentials for this environment are the standard Windows (Active Directory) credentials granted to all INL employees and contractors.

The production environment sits outside the INL firewall in an area referred to as the demilitarized zone (DMZ). The DMZ is a secured area on INL's network that is available to both external (to INL) and internal users, which is managed by INL's Information Management (IM) group. This system is where users access Web pages that present static and dynamic data developed by NDMAS analysts and developers. User credentials for this environment are managed in a separate Active Directory domain. The process for gaining access to the DMZ is managed by IMs Computer Account Management.

The production environment is available to any approved user, including U.S. Citizens and foreign nationals regardless of whether they are INL employees or not. Users are approved based on a need to know. The VHTR TDO Director has ultimate authority in approving users.

7.2 Users Groups and Roles

SAS controls access to data and functionality through a combination of roles, groups, and access control templates. Data, Web page, and functionality rights are granted only as needed and required. Groups are the most important component of managing access permissions to NDMAS data and Web pages.

7.2.1 Roles

Standard SAS roles are used to provide functional access capabilities in SAS software products. Roles for NDMAS are primarily used to grant capabilities to administer the system-to-system administrators. Standard users by default do not have any Role based access granted.

7.2.2 Groups

Groups are used to grant access permission to data and Web pages. By assigning people or assets to a group, the same permissions can be simultaneously assigned to multiple people or Web pages with the same access requirements. There are two categories of groups allowing access rights in a many-to-many way (groups-to-Web pages):

A. Groups for Web pages and portlets

B. Groups for people and data.

7.2.3 Access Control Templates

Access Control Templates provide permission templates to folders in the SAS metadata repository. These templates can then be assigned to groups, facilitating the assignment of access permissions. NDMAS does not use access control templates at this time.

7.3 Group Access Permissions

Access permissions to data and Web pages are managed through the use of groups. Both people and Web pages are assigned to groups. Access permission is then assigned between people-based groups and Web-page-based groups. This makes management of access permissions more efficient. A new user can be assigned to the appropriate people-based group, and will automatically have the necessary access permissions without having to grant permissions Web page by Web page.

People are assigned to groups based on the projects and data streams on which they are working. People groups can be hierarchical. Figure 13 illustrates people-based hierarchical groups for the High Temperature Materials (HTM) project. Some users are limited to access data from a single set of experiments, such as the HTM RPV data stream (User 2). Senior project members have access to data from all of the experiments (User 1). The HTM NDMAS data stream manager has access rights to all elements of the HTM project as well as many other elements related to Web page development and data management that none of the HTM project team have access to. Groups of people all start with ‘Team.’ Groups of users contain the word ‘Customers’ in the group title. A list of current groups defined in the NDMAS system is presented in Table 9.

Table 9. Security groups defined for NDMAS administrators, developers, end users and Web pages.

Group	Description
SAS Administrators	Users who perform metadata administrative tasks.
NGNP Page AGR AGR-1	
NGNP Page AGR AGR-2	
NGNP Page Graphite AGC-1	
NGNP Page Graphite AGC-2	
NGNP Page Hierarchical Web Auto	All User Groups have access to this page launch page. (Not developers.)
NGNP Page Hierarchical Web Nonauto	Developers Only
NGNP Page HTM	
NGNP Page HTM RPV	
NGNP Page HTM SGIHX	
NGNP Page HTM SGIHX Creep	
NGNP Page Qualification	
Team NGNP Customers AGR	Initially for Blaise Collin to access the external Website (a full time BEA employee and a Foreign National). Full access to AGR1 and AGR2.
Team NGNP Customers FR	Limited to the test accounts for the NGNP Customers who are French
Team NGNP Customers SA	Limited to the test group of NGNP Customers who are South African nationals.
Team NGNP Customers US	These users include NDMAS members, the approved list to see all AGR-2 plus all experiments/data streams.
Team NGNP Customers US Limited	These users can see: AGR-1 – All, AGR-2 excluding capsules 1 (French fuel) and 4 (South African fuel)
Team NGNP Customers Graphite	Customers of AGC-x and Baseline Graphite projects
Team NGNP Customers HTM	High Temperature Materials
Team NGNP Customers HTM RPV	High Temperature Materials – Reactor Pressure Vessel
Team NGNP Customers HTM SGIHX	High Temperature Materials – Steam Generator and Intermediate Heat Exchanger
Team NGNP Customers HTM SGIHX Creep	High Temperature Materials – Steam Generator and Intermediate Heat Exchanger – Creep Tests for Alloys 617 and 800H
Team NGNP Customers Preliminary	
Team NGNP Customers Qualification	Qualification – Top Level
Team NGNP Design Methods	
Team NGNP Developers	
Team SQL NDMAS2	
Team SQL NDMAS2 DEV	

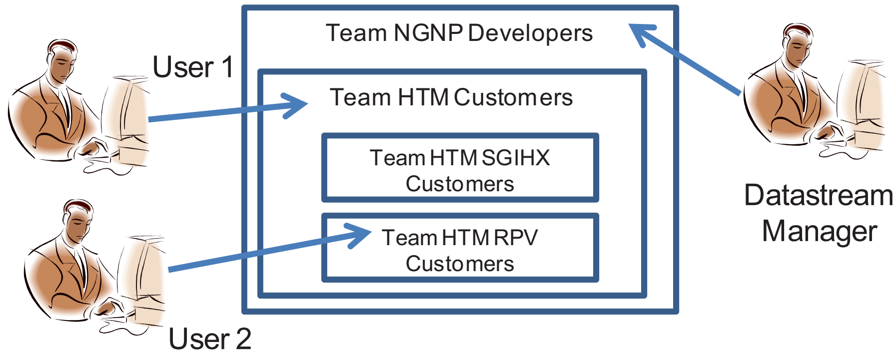


Figure 13. Users are assigned to groups that are granted various levels of access to data and objects.

Web pages and portlets developed by the NDMAS team are also assigned to groups. Web page groups contain the word ‘Page’ in the title. Access to Web pages is then granted by providing access permission between a people-based group and a Web-page-based group as shown in Figure 14. Some Web pages are open to many groups, such as introductory pages. Some customer groups have access to a large number of Web pages. Other customer groups can only access a limited number of pages with data relevant to that particular group. As can be seen from Figure 14, there are many associations to be made between people and Web pages. If these associations had to be made for every member of the VHTR program, it would be a difficult task to manage. But by creating the customer groups, adding people to the project is relatively simple and merely requires assigning them to the appropriate people-based group. Similarly, new Web pages can be easily made available to all appropriate end users by assigning the Web page to the appropriate page group. In Figure 14, multiple HTM SGIHX Web pages are assigned to the Page HTM SGIHX group. All of these pages obtain their access permissions from that group.

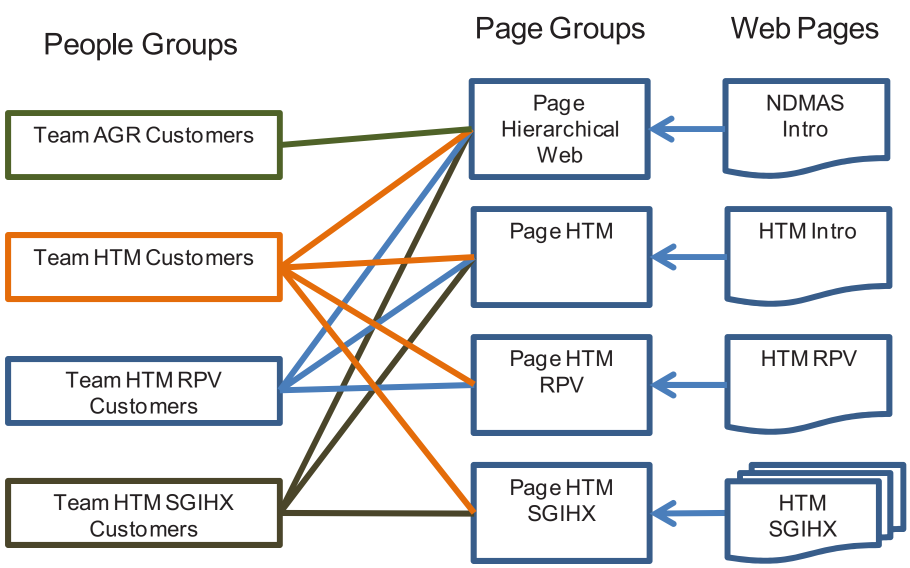


Figure 14. Web pages are assigned to page groups. Access is granted by assigned permissions between people groups and Web page groups.

One responsibility of the data stream manager in establishing a new data stream is to determine the access rights needed by the owners and users of the data stream. New customer and page groups may need to be established for a new data stream. The data generator is responsible for identifying access limitations to data, but the data stream manager is responsible for making sure that Web pages and data are only available to those with proper authorization.

7.4 Data Access Permissions

Sometimes it is necessary to limit access to data with even more granularity than the Web page or portlet. For example, fuel tested in the AGR-2 experiment came from the United States, South Africa, and France. End users from the United States could see irradiation monitoring data gathered from all capsules. French and South African users could see data from the U.S. capsules and their own capsule, but not data from the other foreign country. Access permissions could have been implemented by creating multiple Web pages with different page groups. However, SAS provides tools for controlling data access at row and member levels within individual data tables. Row level permissions are implemented using SAS Information Map Studio as discussed in Section 6.2.4. Member level permissions are implemented using SAS OLAP Cube Studio as discussed in Section 6.2.3. Row and member level permissions are sufficiently complicated to require direct reference to the SAS Security Administration Manual (SAS Institute Inc. 2009).

8. QUALITY ASSURANCE

This section briefly presents the requirements imposed on NDMAS though program plans and INL procedures. Other documents provide details on NDMAS implementation of these requirements.

8.1.1 Quality Assurance

NDMAS operates within the framework of NQA-1 Part I: “Quality Assurance Requirements for Nuclear Facility Applications,” as implemented through PLN-2690, “VHTR Technology Development Office Quality Assurance Program Plan.” NDMAS is a satellite file location as defined in PLN-3319, “Records Management Plan for the VHTR Technology Development Office Program,” which operates within the conditions of that plan. A description of how NDMAS implements the QA and record management requirements is given in PLN-2709, “VHTR Program Data Management and Analysis Plan.”

NDMAS uses software to manage and deliver data. Software is managed following MCP-3058, “VHTR TDO Software Quality Assurance.” The NDMAS Software QA Plan is stored on the NDMAS SharePoint Software Quality Assurance Web page in the NDMAS collaboration environment.

8.1.2 Data Qualification

Data qualification is the act of reviewing, inspecting, testing, checking, or otherwise determining and documenting that data conform to requirements for a specific end use. Company procedure MCP-2691, “Data Qualification,” covers evaluating VHTR Program data for all types of data that require qualification for design or licensing applications. Data conditions may require different approaches to qualification. The data generator is responsible for qualifying data. However, NDMAS can contribute to data qualification. For some data streams, NDMAS may prepare draft data qualification reports for review by the data generators. Data qualification states are:

- *Unevaluated*. Data that have not been evaluated, which covers *Type B* and *Type D* data (see Section 2.1.1 for a discussion of data types).
- *Indeterminate Quality*. Sources of data with a non-NQA-1 QA program or sources with an unknown QA program; *Type C* and *Type D* data would fall into this classification. A quality background is developed for *Type C* data, following MCP-2691 to qualify data. *Type D* data remain in this category.
- *Failed*. Data evaluated for the purposes of qualification that failed to meet requirements. Data do not provide information about the system or object. Data are not useable by the program.
- *Trend*. Data that may not meet all requirements for *Type A* or *Type C* data, yet, because of the data set attributes and/or content, provide some value for an intended use.
- *Qualified*. Data that have been collected and managed under an approved NQA-1 program (*Type A*). Or, data that have been gathered and managed under requirements that meet the intent of an NQA-1 program and have been reviewed, verified, and documented by a data evaluation team that the data meet the requirements for a specific end use as defined in a data collection plan (*Type C*). Any nonconformances are concluded to not affect the usability of the data.

MCP-2691 defines the term ‘Indeterminate Quality’ as data collected outside of a known QA program. NDMAS does not use this category in managing data, but labels this type of data *Unevaluated*. NDMAS uses one additional category (*In Process*), which is applied to data when first received by NDMAS. *Type A* and *Type C* data retain this flag until results of data qualification are final and documented. *Type B* and *Type D* data retain the status of *Unevaluated* after upload to the data warehouse.

NDMAS is responsible for attaching data qualification flags to data in NDMAS to convey the state of the data to the end user.

9. REFERENCES

Gentillon, C. D., M. L. Abbott, L. C. Hull, B. Pham and M. A. Plummer, 2011, *NGNP Data Management and Analysis System Analysis and Web Delivery Capabilities*, INL/EXT-09-16327, Rev 2, Idaho National Laboratory, 49 p.

MCP-2691, "Data Qualification," Idaho National Laboratory, Idaho Falls, ID, June 2011, 30 p.

MCP-3058, "VHTR TDO Software Quality Assurance," Idaho National Laboratory, Idaho Falls, ID, May 2012, 45 p.

PLN-2494, "VHTR Technology Development Office Program Management Plan," Idaho National Laboratory, Idaho Falls, ID, Feb 2012, 60 p.

PLN-2690, "VHTR Technology Development Office Quality Assurance Program Plan," Idaho National Laboratory, Idaho Falls, ID, July 2011, 46 p.

PLN-2709, "VHTR Program Data Management and Analysis Plan," Idaho National Laboratory, Idaho Falls, ID, July 2011, 30 p.

PLN-2803, "Next Generation Nuclear Plant Reactor Pressure Vessel Materials Research and Development Plan," Rev. 1, Idaho National Laboratory, Idaho Falls, ID, July 2010, 231 p.

PLN-3319, "Records Management Plan for the VHTR Technology Development Office Program" Idaho National Laboratory, Idaho Falls, ID, June 2010, 12 p.

Sagi, I., 2011, *Sharepoint 2010*, Sams Publishing, Indianapolis, IN.

SAS Institute Inc., 2009, *SAS® 9.2 Intelligence Platform: Security Adminstration Guide*, SAS Institute inc., Cary, NC.

SDD-228, "NDMAS 2.0 Database," Idaho National Laboratory, Idaho Falls, ID, September 2011, 57 p.

TFR-747, "RDAS-CDCS Data Transfer to NDMAS," Idaho National Laboratory, Idaho Falls, ID, December 2011, 20 p.